

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО

*Факультет інформатики та обчислювальної техніки*

(назва факультету, інституту)

*Кафедра автоматизованих систем обробки інформації і управління*

(назва кафедри)

"На правах рукопису"

УДК 004.023

«До захисту допущено»

Завідувач кафедри

О.А.Павлов

(підпис)

(ініціали, прізвище)

«      »        20 18 р.

**МАГІСТЕРСЬКА ДИСЕРТАЦІЯ**

**на здобуття ступеня магістра**

за спеціальністю 126 Інформаційні системи та технології

(код та назва спеціальності)

ОПП

Інформаційні управляючі системи та технології

(код та назва спеціалізації)

на тему: Інформаційна система пошуку та відбору персоналу

з використанням нейронних мереж

Виконав: студент

VI курсу

ІС-371мп

(шифр групи)

Талько Юрій Сергійович

(прізвище, ім'я, по батькові)

(підпис)

Науковий керівник

проф., д.т.н., доц. Стеценко І.В.

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Консультант

к.т.н., доц. Жданова О.Г.

(науковий ступінь, вчене звання, прізвище, ініціали)

(підпис)

Рецензент

(посада, науковий ступінь, вчене звання, прізвище та ініціали)

(підпис)

Засвідчую, що у цій магістерській дисертації  
немає запозичень з праць інших авторів без  
відповідних посилань.

Студент

(підпис)

Київ – 2018

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**  
**«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. ІГОРЯ СІКОРСЬКОГО»**

Факультет інформатики та обчислювальної техніки  
(повна назва)

Кафедра автоматизованих систем обробки інформації та управління  
(повна назва)

Рівень вищої освіти другий (магістерський) за освітньо-професійною програмою

Спеціальність 126 Інформаційні системи та технології  
(код і назва)

ОПП Інформаційні управляючі системи та технології  
(код і назва)

ЗАТВЕРДЖУЮ  
 Завідувач кафедри  
 \_\_\_\_\_ О.А.Павлов  
(підпис) (ініціали, прізвище)  
 «\_\_» грудня 2018 р.

## ЗАВДАННЯ

### на магістерську дисертацію студенту

Талька Юрія Сергійовича

(прізвище, ім'я, по батькові)

1. Тема дисертації Інформаційна система пошуку та відбору персоналу з використанням нейронних мереж

науковий керівник дисертації д.т.н. доцент Стеценко Інна Вячеславівна  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “ 07 ” листопада 20 18 р. № 4112с

2. Строк подання студентом дисертації “ 3 ” грудня 20 18 р.

3. Об'єкт дослідження процес пошуку та відбору персоналу

4. Перелік завдань, які потрібно розробити 1) Отримати правдоподібні дані про тих хто шукає роботу та рекрутерів 2) Використати методи машинного навчання для фільтрації та підбору вакансій 3) модифікувати метод навчання нейронної мережі з використанням методу дистилляції знань та методу teacher-student 4) Використати модифікацію методу для імітації навчання мережі від декількох teachers (вчителів)

5. Орієнтовний перелік ілюстративного матеріалу 1) Діаграма розгортання;

2) Діаграма варіантів використання; 3) Діаграма діяльності 4) ER-діаграма

5) Діаграма класів; 6) Діаграма компонентів використання

6. Орієнтовний перелік публікацій 1) Талько Ю.С. Методи стиснення моделей в

глибинному навчанні / Міжнародний науковий журнал «Управляющие системы и машины»

УСиМ. – 2018. 2) Талько Ю.С. Методи стиснення моделей в глибинному навчанні /

Всеукраїнська науково-практична конференція молодих вчених та студентів

«Інформаційні / ІСТУ. – 2018 - м. Київ.: НТУУ «КПІ ім. Ігоря Сікорського», 29-30 грудня

2018 р.

## 7. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

8. Дата видачі завдання

“ 29 ” жовтня 20 18 р.

## Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1	Систематизація результатів огляду літератури		
2	Порівняльний аналіз існуючих методів розв'язання задачі		
3	Постановка та формалізація математичної моделі задачі		
4	Модифікація існуючих методів розв'язання задачі		
5	Розробка інформаційного та програмного забезпечення		
7	Проведення експериментальних досліджень розроблених алгоритмів		
8	Оформлення документації		
9	Подання роботи на попередній захист		
10	Подання роботи на основний захист		

Студент

\_\_\_\_\_  
(підпис)

Ю.С. Талько

(ініціали, прізвище)

Науковий керівник

\_\_\_\_\_  
(підпис)

І.В. Стеценко

(ініціали, прізвище)

## РЕФЕРАТ

Магістерська дисертація: 107 с., 19 рис., 33 табл., 7 додатків, 13 джерел.

**Актуальність.** Для будь якої сучасної компанії для виконання своєї прямої задачі важливим є те хто ці задачі виконуватиме. При постійному розвитку, збільшенню організації як в кількісних так і в якісних показниках постає питання підбору якісних кадрів на нові робочі місця. Якщо у випадку з невеличкою компанією її засновник чи менеджер з кадрів котрий був в команді з її зародження може досить ефективно опиратися на власні судження і знання справи якою займається, то при рекрутингу людей до великих корпорацій все стає складнішим. Обов'язки підбору персоналу у різні відділи чи навіть офіси вже неможливо осилити декількома людьми які точно знають якого працівника шукати і якими рисами він повинен володіти. Ці обов'язки делегуються іншим людям, а ті делегують іншим і т.д. Це є нормальним процесом росту організації. Але це також і впливає на якість підбору кадрів. Будь то відсутність контролю за рекрутерами, некомпетентність чи банальні людські помилки – з цим усім компанія стикається у процесі росту.

Одним з вирішенням такого роду проблем є часткова автоматизація процесу підбору кандидатів і вакансій. Адже окрім звичайної каталогізації шукачів та вакансій, котру надають вже існуючі системи, система може попередньо фільтрувати подані кандидатами резюме чи підбирати їм найвідповіднішу навичкам вакансію. Звісно останнім має рішення людини у цьому процесі але система може суттєво зменшити об'єми роботи які потрібно виконати для його прийняття

**Зв'язок роботи з науковими програмами, планами, темами.** Робота виконувалась на кафедрі автоматизованих систем обробки інформації та управління Національного технічного університету України «Київський політехнічний інститут ім. Ігоря Сікорського» в рамках теми «Інтелектуальні методи програмування, моделювання і прогнозування з використанням ймовірнісного і лінгвістичних підходів. Державний реєстраційний номер 0117U000926».

**Мета дослідження** – пришвидшення процесу підбору персоналу в компанії та попередня рекомендація шукачу найкращої вакансії за допомогою використання методів машинного навчання та нейронних мереж.

Для досягнення мети необхідно виконати наступні задачі:

- отримати правдоподібні поточні дані про тих хто шукає роботу;
- отримати правдоподібні дані про тих кого шукають рекрутери.
- використати методи машинного навчання для фільтрації та підбору вакансії.
- модифікувати метод навчання нейронної мережі з використанням методу дистиляції знань та методу teacher-student.
- використати модифікацію методу для імітації навчання мережі від декількох teachers (вчителів).

**Об’єкт дослідження** – процес пошуку та відбору персоналу.

**Предмет дослідження** – методи машинного навчання та нейронні мережу а також способи їх навчання для автоматичного підбору вакансій кандидатам.

**Наукова новизна отриманих результатів** полягає в використанні підходу дистиляції знань і використанні методу навчання нейронної мережі teacher-student їх модифікації у вигляді регуляризатора гаусівського шуму та виборі архітектури нейромережі в контексті практичної задач котра вданому випадку вважається системою пошуку та підбору персоналу.

**Публікації.**

Только Ю.С. Методи стиснення моделей в глибинному навчанні // УСиМ. – 2018.

Только Ю.С. Методи стиснення моделей в глибинному навчанні // ІСТУ. – 2018. – С. 152-156

МОДЕЛЬ, ДИСТИЛЯЦІЯ ЗНАНЬ, ГАУСІВ ШУМ, ГЛИБИННЕ НАВЧАННЯ, НЕЙРОМЕРЕЖІ, НЕЙРОН

## ABSTRACT

Master's dissertation: 107 with., 19 Fig., 33 tabl., 7 applications, 13 sources.

**Topicality.** The high results of using the approach of deep learning of models led to the creation of models with greater depth which, in turn, allowed to solve more complex problems. On the other hand, it also leads to an increase in the depth of the models, which increases complexity and reduces the execution time. To solve this delivery, it is fashionable to use compression of model volumes.

**Relationship of work with scientific programs, plans, themes.** The work was carried out at the Department of Automated Systems for Information Processing and Management of the National Technical University of Ukraine "Kyiv Polytechnic Institute. Igor Sikorsky" within the theme « Intelligent methods of programming, modeling and forecasting using probabilistic and linguistic approaches. State registration number 0117U000926»

**The object of the research** - the process of recruiting.

**Subject of research** - methods and models for the formation of personal recommendations.

**The research methods** methods of machine learning and neural network, as well as ways of teaching them for the automatic selection of vacancies for candidates.

**The scientific novelty of the results** is use the distillation of knowledge method and the use of the teacher-student for teaching neural network and modify them in the form of a regularizer of Gaussian noise and the choice of the architecture of the neural network in the context of practical tasks,

### **Publications**

Talko Y.S. Compression methods of deep learning models based on teacher-student method [Electronic resource] // usim. - 2018.

Talko Y.S. Compression methods of deep learning models based on teacher-student method [Electronic resource] // ISMT. - 2018. – pp 152-156:

MODEL, DISTILLATION OF KNOWLEDGE, GAUSSIAN NOISE, DEEP LEARNING, NEURAL NETWORKS, NEURON

## ЗМІСТ

1	ПРОЕКТНІ РІШЕННЯ З РОЗРОБКИ ПІДТРИМКА ПРОЦЕСУ РЕКРУТИНГУ В ІНФОРМАЦІЙНИХ СИСТЕМАХ .....	8
1.1	Опис бізнес-процесів .....	8
1.1.1	Опис процесу діяльності .....	8
1.1.2	Актори і функції.....	11
1.1.3	Структура бізнес-процесів .....	17
1.2	Опис постановки задачі .....	19
1.3	Огляд аналогів системи, що розробляється .....	20
1.4	Рішення з інформаційного забезпечення системи пошуку та відбору кандидатів	22
2	МОДЕЛІ ТА МЕТОДИ ВИКОРИСТАННЯМ НЕЙРОННИХ МЕРЖ В СИСТЕМІ ПОШУКУ ТА ПІДБОРУ ПЕРСОНАЛУ.....	28
2.1	Змістовна постановка задачі .....	28
2.1.1	Задача підбору персоналу .....	28
2.1.2	Використання алгоритмів машинного навчання .....	32
2.2	Математична модель оцінки придатності кандидата.....	36
2.2.1	Задача оцінки придатності кандидата.....	36
2.2.2	Задача класифікації даних за набором критеріїв.....	37
2.3	Огляд методів розв'язання задачі класифікації .....	38
2.3.1	Метод опорних векторів.....	39
2.3.2	Використання підходу дистиляції знань .....	44
2.4	Модифікація методу вчителя-ученика.....	45
2.5	Результати досліджень ефективності методу.....	47
3	ОПИС ПРОГРАМНОГО ТА ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ .....	52

3.1	Засоби розробки .....	52
3.2	Архітектура програмного забезпечення .....	53
3.2.1	Загальні положення.....	53
3.3	Настанова користувача.....	58
4	РОЗРОБКА СТАРТАП-ПРОЕКТУ .....	69
4.1	Опис ідеї проекту (товару, послуги, технології).....	69
4.1.1	Аналіз потенційних техніко-економічних переваг ідеї .....	70
4.2	Технологічний аудит ідеї проекту.....	75
4.3	Аналіз ринкових можливостей запуску стартап-проекту.....	77
4.4	Розроблення ринкової стратегії проекту .....	87
4.5	Розроблення маркетингової програми стартап-проекту .....	92
	ВИСНОВКИ.....	97
	ПЕРЕЛІК ПОСИЛАНЬ .....	99
	ДОДАТОК А Графічний матеріал.....	100
	Плакат 1. Діаграма варіантів використання .....	101
	Плакат 2. Діаграма розгортання .....	102
	Плакат 3. Діаграма діяльності.....	103
	Плакат 4. ER-діаграма.....	104
	Плакат 5. Діаграма класів.....	105
	Плакат 6. Діаграма компонентів застосування .....	106
	Плакат 7. Діаграма послідовності.....	107



# **1 ПРОЕКТНІ РІШЕННЯ З РОЗРОБКИ ПІДТРИМКА ПРОЦЕСУ РЕКРУТИНГУ В ІНФОРМАЦІЙНИХ СИСТЕМАХ**

## **1.1 Опис бізнес-процесів**

Однією з найважливіших задач яка стоїть перед компанією як в будь який період її існування є підбір кваліфікованих працівників. Сьогодні в кожній компанії є від 1-2 рекрутерів до цілих окремих відділів рекрутингу, працівники котрих за день можуть провести десятки співбесід - вдалих чи ні. Основне рішення при наймі працівника завжди має залишатися за людиною, але все одно існує багато інструментів які так чи інакше полегшують чи пришвидшують цю роботу. Ці інструменти виконують багато задач, як наприклад упорядковування, виставлення пріоритетів, ведення бази потенційних кандидатів і т.д. Однак одним з найбільш ефективних з точки зору оптимізації робочого процесу рекрутерів бізнес процесів, є сама співбесіда. І враховуючи велику кількість етапів співбесід навіть для однієї людини, це є дуже затратною дією. А для того щоб цей бізнес процес оптимізувати, мабуть найкращим рішенням буде зменшити кількість співбесід за допомогою інформаційної системи яка зможе суттєво зменшити кількість опитуваних кандидатів.

### ***1.1.1 Опис процесу діяльності***

Рекрутинг — набір кандидатів до робочого штату компанії, або на замовлення клієнта якщо пошук відбувається через рекрутингове агенство; головний функціональний обов'язок менеджерів по персоналу (hr) та рекрутерів.

Для рекрутингу використовують різні джерела інформації про кандидатів а також різні підходи до їх пошуку в них. Основні джерела можна показати списком наведеним нижче:

- база даних компанії котра шукає;
- сайти котрі спеціалізуються на пошуку роботи;
- соціальні зв'язки (пошук кандидатів по знайомству);
- ЗМІ
- соцмережі, спеціалізовані форуми;

- переманювання спеціалістів з компаній-конкурентів;
- залучення нових спеціалістів з навчальних закладів;

Рекрутинг можна умовно поділити на кілька видів:

- Класичний рекрутинг – пошук спеціаліста без обмежень джерела інформації чи впливу на рішення кандидата.

- Масовий набір персоналу – здійснюється коли відкривається нова компанія чи філії вже існуючої компанії. Прикладом може послужити відкриття нового супермаркету. Вартість підбору кандидата за такого типу набору значно нижча ніж при наступному типі рекрутинг.

- Хедхантинг — переманювання важливого спеціаліста у компанії-конкурента. Ціна подібної послуги у рекрутингових агенствах є найбільшою.

- Хедхантинг вважається нечесним методом набору персоналу але є зазвичай, найефективнішим, так як спрямований на найуспішніші кадри компанії. Адже витратити таку кількість ресурсів для рекрутингу є сенс лише у випадку коли точно відомо про цінність кадру.

- Масовий набір персоналу зазвичай є найменш ефективним, так як при великій кількості вакансій збільшується також і кількість кандидатів. З чого випливає що зосередитись на підборі якісного персоналу відділ рекрутингу чи рекрутингова компанія просто не може або ж це недоцільно через замалу кількість часу виділену на пошук персоналу. Для рекрутерів проводити занадто жорсткий чи якісний підбір недоцільно тому що кількість ресурсів витрачених, на дійсно якісний у стислі строки, підбір кадрів часто недоцільно через обмеження ресурсів. Такий вид рекрутингу, як було згадано вище, прийнято використовувати лише при відкритті нової компанії. В іншому ж випадку проводити масовий рекрутинг недоцільно з огляду на співвідношення отриманого результату і витрачених ресурсів

- Класичний же набір є найбільш надійним та найменш затратним. Саме тому рекрутингові компанії найжвавіше беруться за такий тип залучення нових кадрів. Хоча хедхантинг чи масовий набір, на перший погляд, мають приносити більший прибуток але несуть в собі і найбільші ризики. Скажімо для вдалого хедхантингу

потрібно мати достатньо інформації для того щоб переманити співробітника іншої компанії. Якщо цієї інформації немає то її потрібно здобути. Для того щоб здобути інформацію потрібно витратити певну кількість ресурсів, що лише на етапі підготовки вже знижує чистий прибуток рекрутера. Схожа ситуація з масовим набором, адже для того щоб провести його потрібно мати велику і актуальну базу кандидатів спрямовану на потрібний рід діяльності. Також цьому разі потрібно витратити немало ресурсів на обробку даних великої кількості, що також знижує результуючий прибуток.

- Машинне навчання (англ. machine learning) — це підгалузь інформатики, котра еволюціонувала з інших її підрозділів – з теорії обчислювального навчання та розпізнавання образів в галузі штучного інтелекту.

- Типи задач і завдань

- Задачі машинного навчання, як правило, поділяють на три широкі категорії, в залежності від природи «сигналу», якого навчається система, або «зворотного зв'язку», доступного системі, яка навчається. Цими категоріями є:

- навчання з учителем (кероване навчання, англ. supervised learning): алгоритму надаються певні дані і результати обробки цих даних заданих «вчителем» системи для навчання системи загального правила яке відображає зв'язок між вхідними та вихідними даними;

- навчання без учителя: алгоритму не надається попередніх тренувальних даних. Алгоритм повинен починати вчитися вже на «живих» даних і робити свої висновки. Навчання без учителя може використовуватися для виявлення прихованих закономірностей у даних чи навчанню ознак;

- навчання з підкріпленням (англ. reinforcement learning): алгоритм взаємодіє з динамічним середовищем, у якому він же повинен виконувати певну поставлену мету (наприклад як водіння автомобіля), при цьому вчителя як такого в нього не має, а відповідно дізнатися алгоритм чи підійшов він до поставленої цілі не може. Прикладом можна вважати навчання гри в шахи через гру з живим суперником.

### **1.1.2 Актори і функції**

Для того щоб створити діаграми використання спочатку необхідно позначити дійових осіб(акторів), а після того визначити, які дії може виконувати кожен з акторів. Одним з акторів у системі є адміністратор системи, котрий має змогу редагувати, створювати чи видаляти заявки за своїм бажанням, також може змінювати різноманітні налаштування системи. Також є користувачі, котрі мають змогу реєструватися, створювати резюме та заповнювати анкети для вподобаних вакансій. Останнім актором є рекрутери, які мають змогу реєструвати роботодавців, створювати чи редагувати вакансії, приймати рішення про закриття вакансії і вибору кандидатів які на них підходять.

Нижче наведений опис можливостей кожного з акторів.

**Кандидат.** Кандидат в системі (КС) може виконувати наступні функції:

- пошук та заповнення заявок до вакантних посад;
- пошук та вакантних посад;
- відстеження заповнених заявок (статус заявки).

**Адміністратор.** Адміністратор системи (АС) може виконувати наступні функції:

- створювати, редагувати та видаляти заявки, вакансії, користувачів, організації;

**Рекрутер.** Рекрутер в системі (РС) може виконувати наступні функції:

- створювати, редагувати та видаляти вакансії;
- приймати, відхиляти чи схвалювати надіслані кандидатами заявки.

Тепер ми можемо визначити дії, котрі можуть виконувати актори в системі (рисунок 1.5):

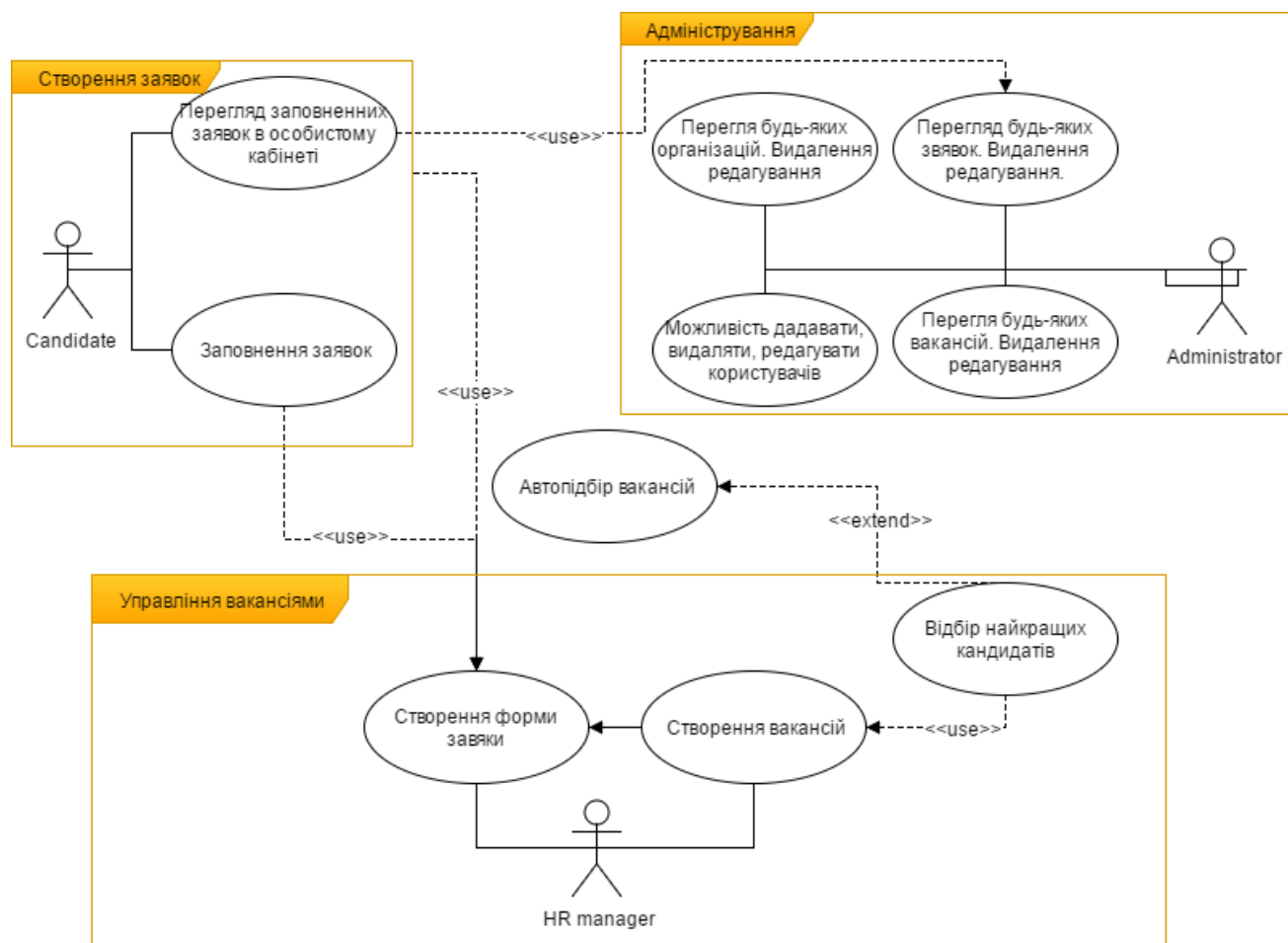


Рисунок 1.1 – Схема структурна варіантів використання

Розкриємо детальніше деякі з варіантів використання в таблиці 1.1:

Таблиця 1.1 – Опис варіантів використання

Актор	Назва варіанту	Опис
Користувач	Перегляд заповнених заявок в особистому кабінеті	Користувач заходить в особистий кабінет і може побачити список активних чи архівних заявок, і їх поточний статус
	Заповнення заявок	Користувач заповнює форму заявки для вибраної вакансії

Актор	Назва варіанту	Опис
Адміністратор	Перегляд будь-яких організацій. Видалення редагування	Адміністратор може додати, видалити чи редагувати організацію.
	Перегляд будь-яких заявок. Видалення редагування.	Адміністратор може додати, видалити чи редагувати заявку
	Можливість додавати, видаляти, редагувати користувачів	Адміністратор може додати, видалити чи редагувати користувача
	Перегляд будь-яких вакансій. Видалення редагування	Адміністратор може додати, видалити чи редагувати вакансію
Рекрутер	Створення форми заявки	Рекрутер створює форму для заповнення кандидатом додатково розставляючи для кожного пункту анкети оцінку важливості
	Створення вакансії	Рекрутер створює та описує вакансію для якої на яку він шукає кандидата
	Вибір найкращого кандидата	Рекрутер вибирає кандидата із запропонованих системою та після серії випробувань

Актор	Назва варіанту	Опис
		затверджує на посаду найкращого кандидата. Вакансія після цього закривається

Також для підвищення надійності системи буде впроваджена мікросервісна архітектура програмного забезпечення. Тобто вся система виноситься на різні фізичні або віртуальні носії, розподіляючись по підсистемам. Така архітектурна побудова є важливою для підтримання постійної працездатності системи. Компоненти на які буде поділена система будуть наведені в таблиці 1.2 нижче. Взаємозв'язки між компонентами будуть зображені на рисунку 1.2

Таблиця 1.2 – Компоненти ситеми

Назва компонента	Призначення
Сервер з веб застосунком	Сервер на якому розміщений фронтенд застосунок. Приймає всі запити користувачів. Може бути декілька для зменшення навантаження на систему у випадку масштабування системи. Представлений веб сервером Node.js та застосунком написаним на React.js
Веб сервер маршрутизації	Основною задачею цього веб-сервера є передавати запити з між аналітичним сервером та фронтенд сервером. Виконує задачу фільтра та «першого рубежу» у випадку атак на систему, фільтрує запити котрі йдуть з зовнішнього світу
Аналітичний веб сервер	На ньому розміщенні основні програмні засоби для обробки інформації та її аналізу.

	Саме тут розташована модель нейромережі та інші програми обробки даних
Сервер реляційної бази даних	Відповідає за збереження постійної інформації (резюме, рекрутери, кандидати і т.д.), також зберігає «знання» які накопичилися завдяки алгоритмам машинного навчання чи були передані з іншого місця
Сервер нереляційної бази даних	Тут зберігається різна тимчасова інформація в тому числі користувацька (сесії).



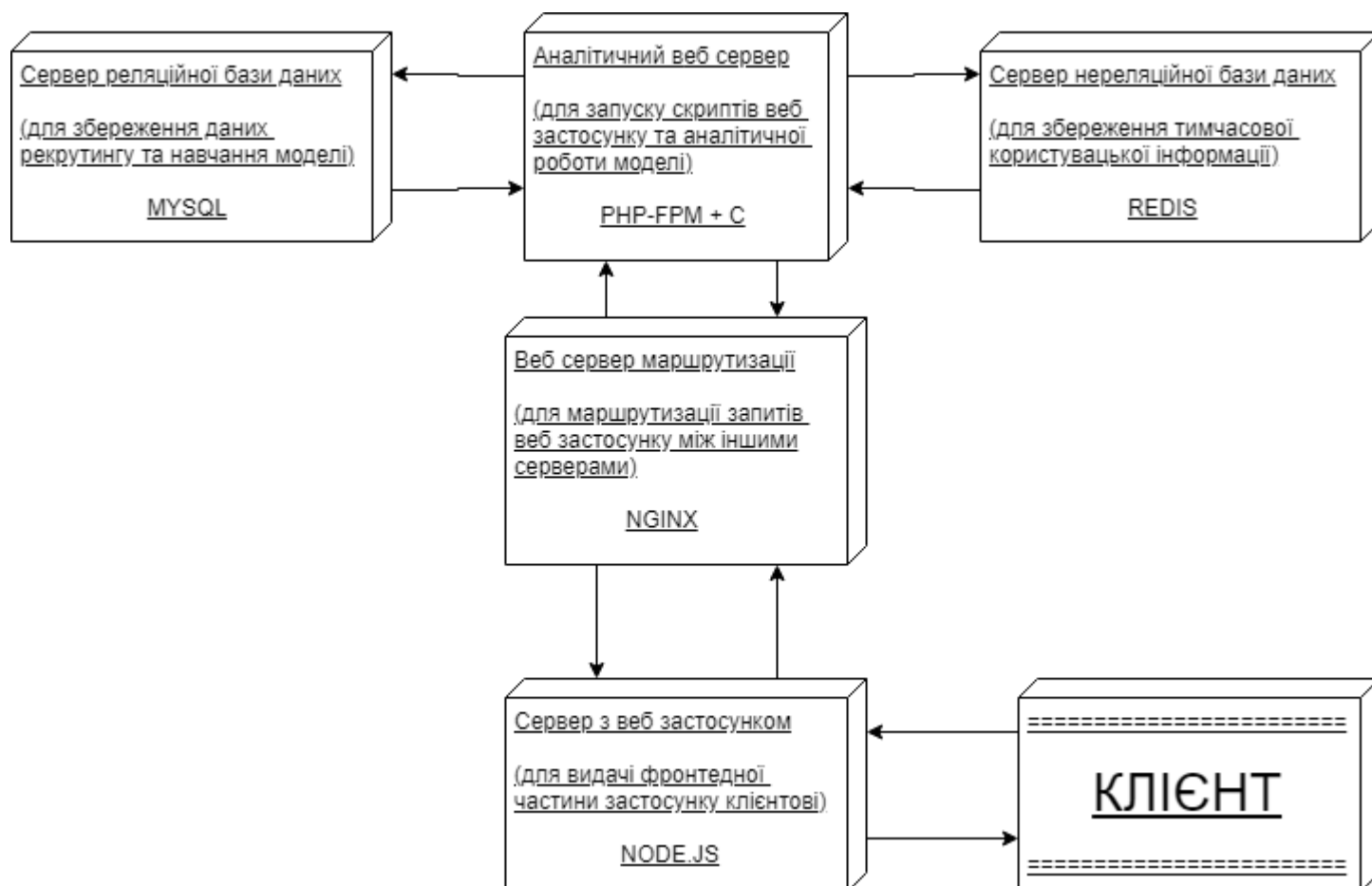


Рисунок 1.2 – діаграма розгортання

### 1.1.3 Структура бізнес-процесів

Дії котрі може виконувати рекрутер щоб створити вакансію зобразимо на UML діаграмі діяльності (на рисунку 1.3).

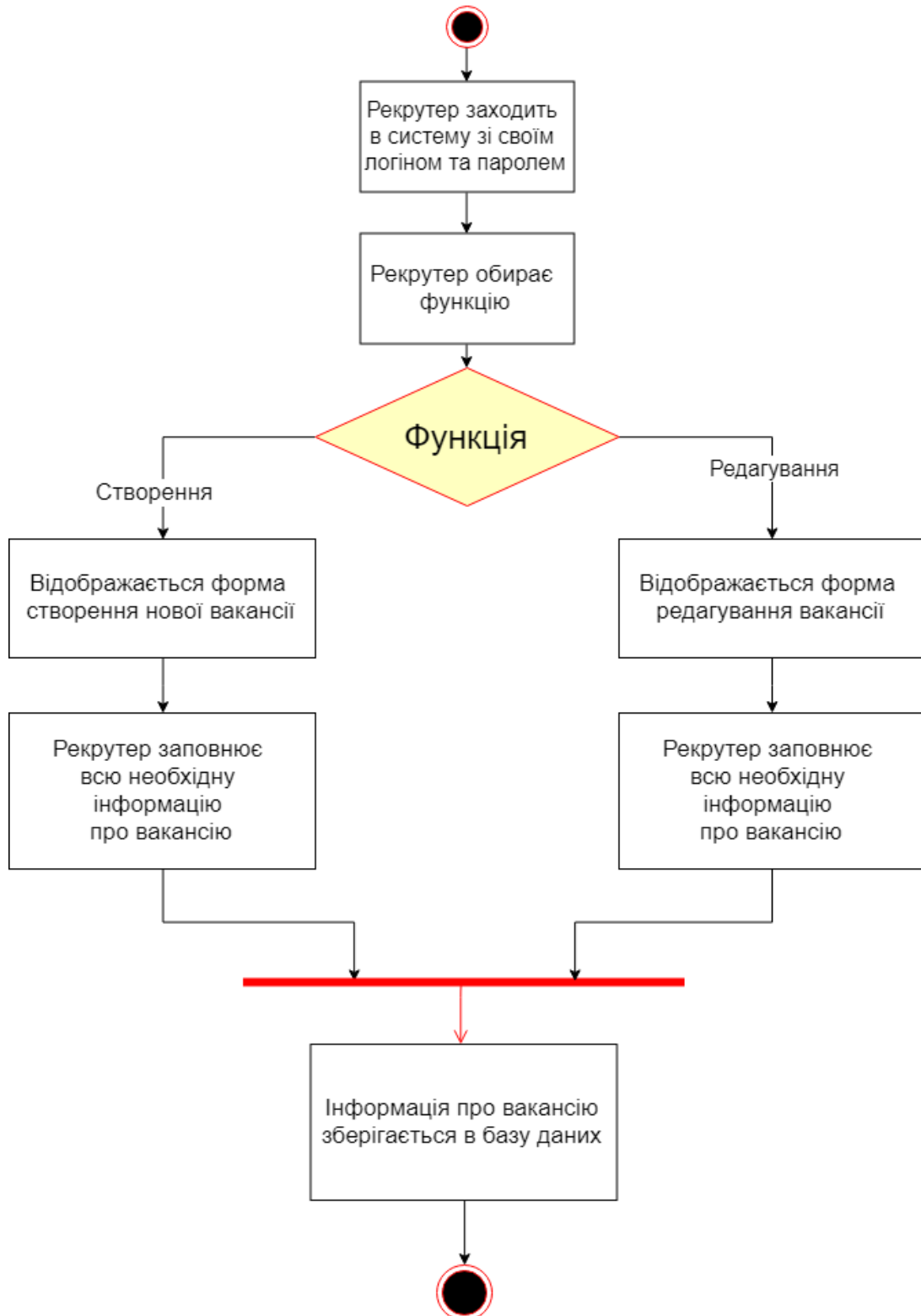


Рисунок 1.3 – Схема структурної діяльності створення/редагування вакансії

Також розглянемо процес створення вакансії рекрутером на рисунку 1.4

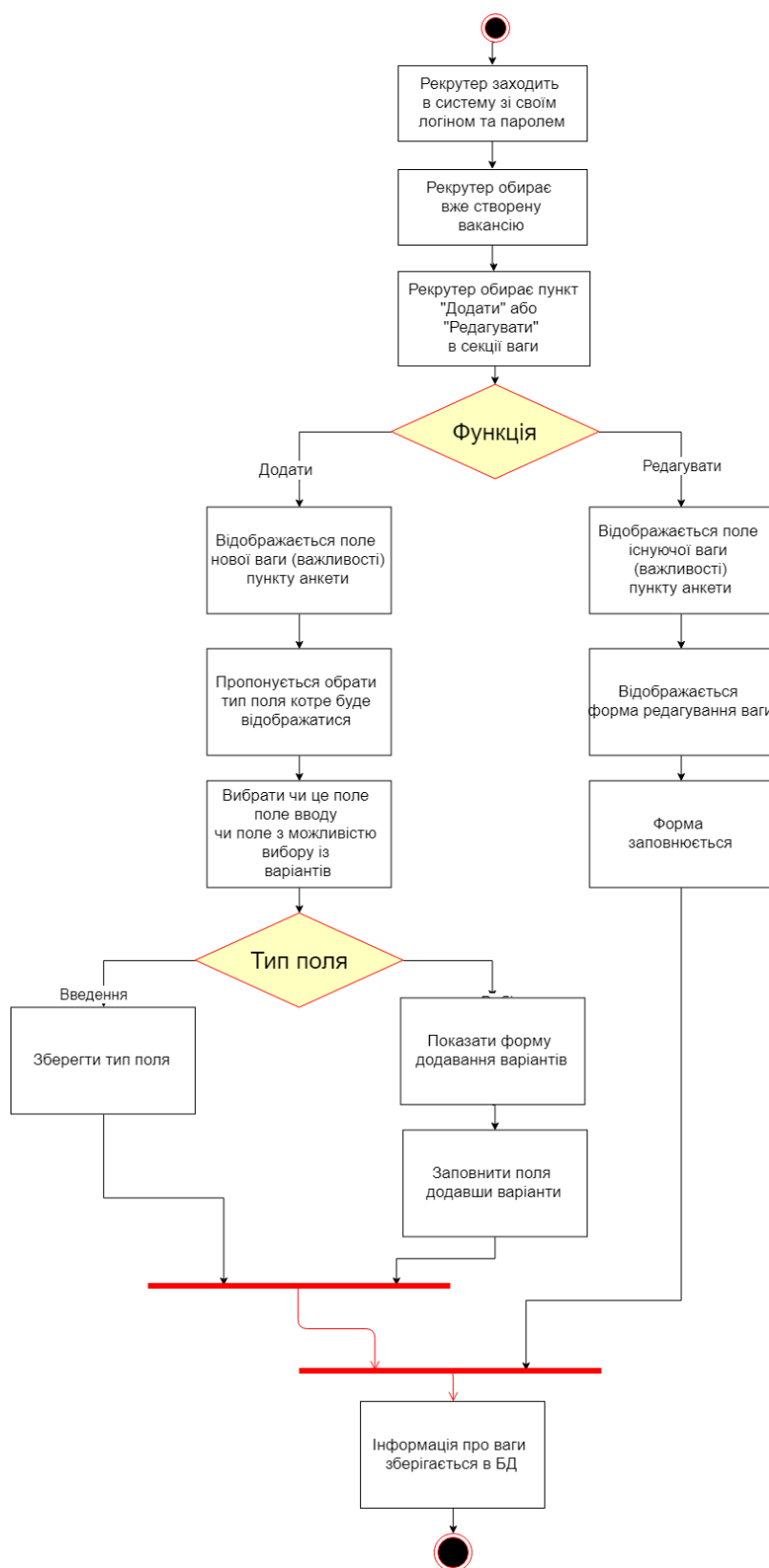


Рисунок 1.4 - схема структурної діяльності створення/редагування вагів вакансії

## 1.2 Опис постановки задачі

Призначенням системи підтримка процесу рекрутингу в інформаційних системах з використанням нейронних мереж є спрощення пошуку вакансій і кандидатів користувачами системи, а також мінімізація людського фактору на підбір найкомпетентніших кандидатів.

Цілі створення системи:

- підвищення ефективності підбору кадрів з мінімізацією людського фактору і урахуванням імовірної подальшої успішності кандидатів.

Для реалізації поставлених цілей система має вирішувати такі задачі:

- пошук найперспективніших кандидатів для вибраної вакансії;
- визначення найбільш підходящих вакансій для кандидата;
- ведення історії індивідуальної для кожного користувача, для накопичення «словника» для машинного навчання.

**Задачі системи:** Для реалізації поставлених цілей система повинна вирішувати такі задачі:

1. Ведення бази кандидатів
  - а. Додавання нових кандидатів
  - б. Редагування інформації про вже існуючих кандидатів
  - с. Видалення інформації про кандидатів
2. Ведення бази вакансій
  - а. Додавання нових вакансій
  - б. Редагування інформації про вже існуючі вакансії
  - с. Видалення інформації про вакансії
3. Автоматичний підбір вакансій
  - а. Навчання та перенавчання моделі
  - б. Визначення найкращої вакансії для кожного окремого кандидата
  - с. Використання підходу дистиляції знань до навчання нейромережі

### 1.3 Огляд аналогів системи, що розробляється

В ході пошуку схожих рішень було виявлено деякі сервіси зі схожою функціональністю:

- «All-in-one recruiting software for ambitious companies»;
- «Оценивайте и сравнивайте кандидатов онлайн»;
- «Talent Management Software with a Clear Mission for Success»;
- Persia автоматизация рекрутинга.

Всі сервіси направлені на взаємодію рекрутерів та кандидатів. Кандидати в деяких підбираються автоматично без прямої взаємодії кандидата за системою на початкових етапах, а деякі потребують бути зареєстрованим кандидатом. В рекрутерів на всіх сайтах роль однакова – вибрати найкращих кандидатів з запропонованих системою.

В таблиці 1.3 наведено короткий опис характерних особливостей існуючих сервісів.

Таблиця 1.3 – Наявні аналоги системи

Назва	Переваги	Недоліки
<a href="https://www.workable.com/">https://www.workable.com/</a>	Потужна платформа що давно на ринку і добре себе зарекомендувала. Надає можливість як кандидатам так і рекрутерам, після реєстрації, шукати перетин інтересів (тобто співпадіння в	Рішення платне.  Не найкраща рекламна компанія. Наприклад, на головній сторінці сайту, звідки починається знайомство з ним, немає інформації про попередні успіхи компанії, а лише загальні відомості та

Назва	Переваги	Недоліки
	зацікавленості рекрутера і кандидата)	багато лозунгів. Сайт є тільки англійською мовою.
<a href="https://gorecru.it/">https://gorecru.it/</a>	Дозволяє рекрутеру шукати кандидатів без їх (кандидатів) прямої в цьому участі, аналізуючи дані соц. мереж потенційних кандидатів. Проста процедура зв'язку рекрутера з кандидатом	Система більше орієнтована на рекрутерів, ніж на кандидатів. Самі кандидати можуть і не знати, що їх шукають через цю систему. Рішення платне. Оплата прив'язана до кількості «оцінок», які можна провести
<a href="http://info.clearcompany.com/">http://info.clearcompany.com/</a>	Потужне рішення, реалізоване відразу на декількох платформах (окрім веб версії, також є версії застосунків для IOS та Android)	Рішення занадто «важке». Гарно підходить для великих компаній з великими HR відділами та великою кількістю ресурсів, для малих же компаній потребує занадто багато витрат (як грошей так і часу). Рішення платне.
<a href="https://www.persiahr.com/ru/">https://www.persiahr.com/ru/</a>	Давно на ринку, знає всі свої переваги і недоліки і користується цим	Технічно застарілий сайт. Рішення платне

### 1.4 Рішення з інформаційного забезпечення системи пошуку та відбору кандидатів

У таблицях 1.4 – 1.9 наведена структура таблиць бази даних.

В таблиці 1.4 наведено опис таблиці candidates, в якій зберігаються кандидати на зайняття вакансії

Таблиця 1.4 – Опис таблиці candidates

Назва	Опис	Тип даних
Pib	Ім'я кандидата	string
birthDate	Дата народження	Date
maritalStatus	Сімейний стан	String
cityId	ID міста проживання	Number
educationId	ID типу освіти кандидата	Number
phoneMobile	Номер мобільного телефону кандидата	string
phoneHome	Номер домашнього телефону кандидата	string
email	Електронна адреса кандидата	String
drivingExp	Досвід водіння	String
recommendations	Рекомендації з попереднього місця роботи	String
changed	Дата зміни запису	Date

Назва	Опис	Тип даних
created	Дата створення запису	Date

В таблиці 1.5 наведено опис таблиці inputs, котра зберігає поля вакансії

Таблиця 1.5 – Опис таблиці inputs

Назва	Опис	Тип даних
column	Номер колонки	string
title	Назва колонки(поля анкети)	string
type	Тип колонки	string

В таблиці cities , котра описана в таблиці 1.6, зберігаються найменування міст в яких можуть проживати кандидати чи бути зареєстровані компанії

Таблиця 1.6 – Опис таблиці cities

Назва	Опис	Тип даних
Country	Країна	string
title	Назва міста	string
coordinates	Координати міста	string

В таблиці 1.7 наведено опис структури таблиці departments яка зберігає компанії котрі рекрутують кандидатів

Таблиця 1.7 – Опис таблиці departments



Назва	Опис	Тип даних
cityId	ID міста	number
title	Назва компанії	string
address	Адреса компанії	string

В таблиці 1.8 наведено опис таблиці бази даних для збереження інформації про вакансію.

Таблиця 1.8 – Опис таблиці vacancies

Назва	Опис	Тип даних
title	Назва вакансії	string
dateEnd	Дата до якої вакансія відкрита	Datetime
categoryId	Тип вакансії	number
departmentId	компанія	number
changed	Коли запис був змінений	Date
created	Коли запис був створений	Date

В таблиці бази даних weights зберігаються ваги важливості кожного поля анкети. Структура таблиці наведена нижче в таблиці 1.9

Таблиця 1.9 – Опис таблиці weights

Назва	Опис	Тип даних
type	Тип поля	string
vacancyId	ID вакансії	number
inputId	ID поля для якого задається вага	Number
value	Значення поля	Number
operation	Операція для пошуку	string

Схема бази даних наведена нижче на рисунку 1.4 в вигляді ER-діаграми

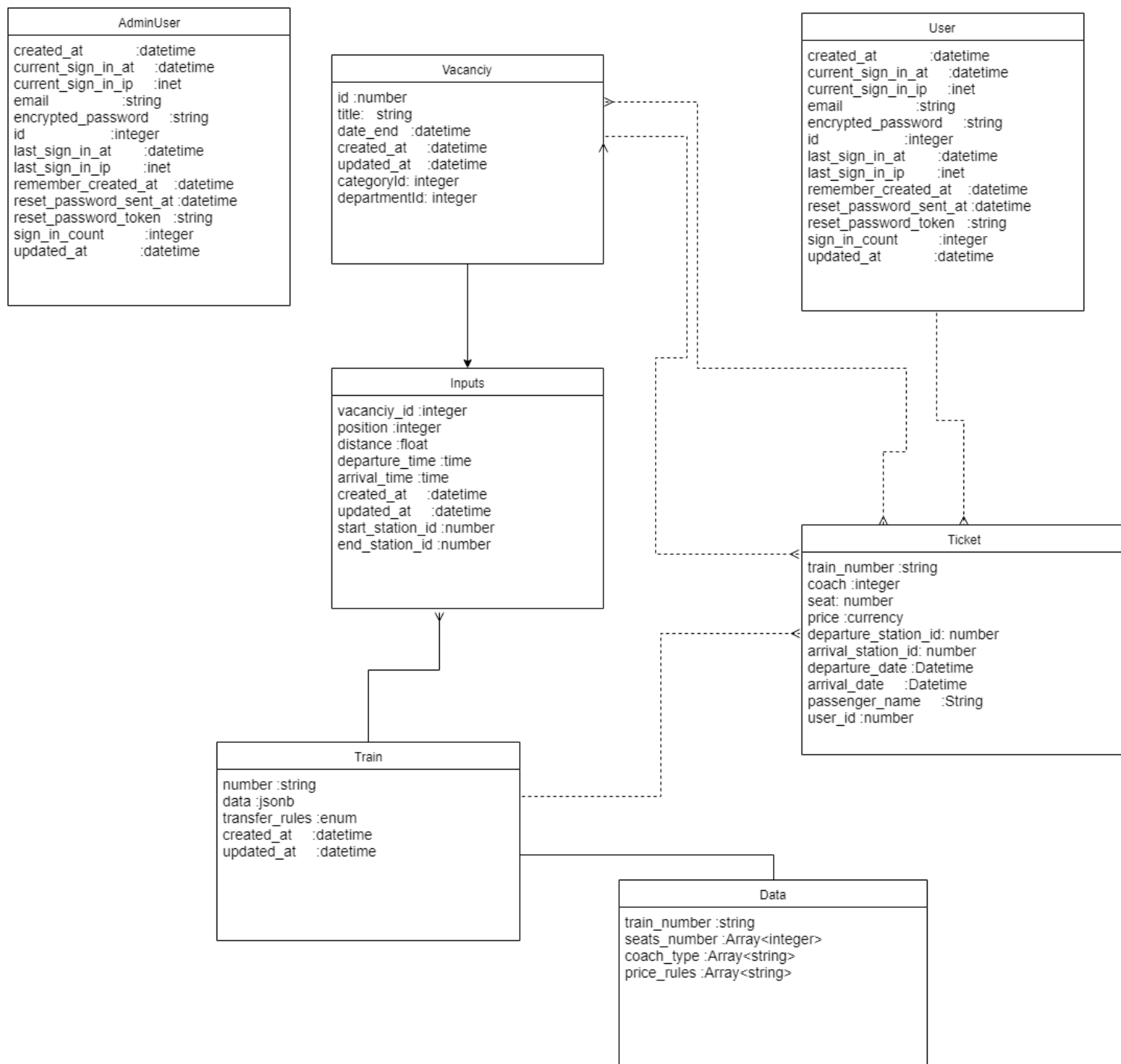


Рисунок 1.4 – ER – діаграма бази даних

## Висновок

Так як основним бізнес-процесом системи є створення вакансій, підбір кандидата під неї, і навпаки рекомендації вакансій тим хто їх шукає було прийнято рішення розробки веб застосунку котрий може виконувати основні задачі наведені в розділі вище так зв'язувати між собою основних акторів системи. Також за допомогою мікросервісної архітектури (яка зручно реалізується як раз за допомогою технологій веб розробки) було підвищена надійність системи і її стійкість до можливих спам атак. Основні задачі системи, окрім зберігання інформації про вакансії і кандидатів в першу чергу, є попередня рекомендація рекрутеру найкращих кандидатів на вакансію і, навпаки, рекомендація найкращих вакансій кандидату. З описаного вище можна зробити висновок, що алгоритми машинного навчання та нейромережі зокрема здатні її вирішувати. Однак, так як одним з мінусів підходу машинного навчання є те що, система потребує відповідно самого навчання, було вирішено розширити звичайний алгоритм ідеєю і моделлю, яка буде описана в наступному розділі.

## 2 МОДЕЛІ ТА МЕТОДИ ВИКОРИСТАННЯ НЕЙРОННИХ МЕРЖ В СИСТЕМІ ПОШУКУ ТА ПІДБОРУ ПЕРСОНАЛУ

### 2.1 Змістовна постановка задачі

#### 2.1.1 *Задача підбору персоналу*

Рекрутинг (підбір персоналу) — процес встановлення відповідності працівника вимогам вакансії шляхом вивчення психологічних і професійних якостей кандидата та підбору з наявних претендентів такого, відповідність якого найбільша. При підборі враховується кваліфікація, спеціальність, особисті якості, здібності, характер кандидата та інтереси організації, котра його наймає. Весь процес є достатньо складним і дорогим. Підбір зазвичай розбивається на декілька етапів. Першим кроком є співбесіда, потім заповнюються різноманітні анкети чи заяви, проводиться окрема співбесіда з менеджером по персоналу, тестування, перевірки наданих рекомендацій, медогляд і т.д. За результатами вище описаних кроків керівники або менеджер з персоналу приймають рішення – кандидат буде найнятий чи ні. Процедура відбору персоналу включає в себе такі етапи:

Мета попереднього відбору – знизити витрати щодо наймання за рахунок скорочення кількості претендентів, що підлягають оцінці. Первинний відбір здійснюється за формально встановленими мінімальними вимогами, що пред'являються підприємством і вакантним робочим місцем до потенційного працівника. Мінімальні вимоги визначаються для кожної конкретної посади у відповідності зі специфікою виконуваних функцій. Такими вимогами можуть бути: досвід, освіта, кваліфікація (наприклад, наявність прав на водіння автотранспортом), навички (володіння ПК і уміння працювати з певними програмами), вік. Попередній відбір здійснюється менеджером з персоналу на основі аналізу резюме.[11]

Резюме являє собою спосіб інформування роботодавця про найбільш важливі характеристики претендента на одержання робочого місця в даному підприємстві. Правильно складене резюме – одна із складових можливого успіху при пошуку роботи. Його мета – привернути увагу, зацікавити роботодавця кандидатурою, представленою в резюме, і спонукати запросити претендента до участі в наступному

відборі. Тому воно повинно містити інформацію про придатність претендента до пропонованої роботи.

При наявності великої кількості претендентів використовується комп'ютерна обробка представлених резюме. Вона заснована на їхньому ранжируванні у залежності від кількості ключових слів (професійних термінів, назв підприємств, навчальних закладів, посад, стажу і досвіду роботи і т.д.) або задоволення певних початкових умов заданих вакансією.

**Проведення первинної співбесіди** – наступна стадія відбору персоналу.

Первинна співбесіда проводиться спеціалістом відділу кадрів. Його мета – більш детальне знайомство з претендентом, що дозволяє визначити його придатність до виконання майбутньої роботи. У процесі співбесіди претендент повинен одержати також інформацію про підприємство і майбутню діяльність, що визначить ступінь його зацікавленості у пропонованій роботі. На цей аспект первинної співбесіди спеціалісти відділу кадрів звертають, як правило, значно менше уваги. Разом з тим, одержання працівником найбільш повної інформації про характер майбутньої діяльності є важливим фактором зниження майбутньої плинності кадрів. Попередня співбесіда є також засобом реклами, що інформує про імідж підприємства. Усе це варто враховувати при підготовці до проведення попередньої співбесіди.

Головна задача менеджера, що проводить інтерв'ю, – одержання аналіз інформації. Тому у кожному інтерв'ю необхідно прагнути до удосконалення:

- техніки формулювання питань для одержання максимального обсягу необхідної інформації;
- критеріїв оцінки отриманої інформації і визначення її значення для прийняття рішення.

Основною метою відбіркової співбесіди (інтерв'ю) є одержання відповіді на питання: чи зацікавлений претендент у даній роботі і чи здатний він її виконувати.

В процесі відбіркової співбесіди необхідно одержати інформацію за наступними блоками:

### **Життєвий шлях, автобіографічні дані**

Особливу увагу необхідно приділити "білим плямам" – періодам часу, коли кандидат з певних причин не навчався або не працював, порівнюючи викладене у резюме (анкеті) із записами у трудовій книжці і документами про освіту.

### **Загальноосвітній рівень**

Визначається: базова освіта, конкретний навчальний заклад, факультет, спеціалізація. Важливо визначити відношення до навчання, ступінь теоретичної підготовки кандидата, тому слід вивчити його дані про підвищення кваліфікації або одержання інших видів освіти, суміжних спеціальностей, проходження курсів і т.п. В даний час курси підвищення кваліфікації можуть тривати від декількох днів до декількох місяців, тому тривалість перепідготовки може відігравати значну роль при визначенні кваліфікації кандидата.

### **Практичний досвід**

Визначається: час і місце роботи на конкретних посадах, зміст виконуваної роботи, функції і обов'язки, можливості виконання інших видів робіт, не передбачених посадовими обов'язками, можливості сполучення професій, спеціальні навички, загальні навички, досвід організаторської роботи або роботи на управлінських посадах. Переходячи до обговорення конкретної вакансії, варто поцікавитися у кандидата, як він розуміє зміст майбутньої роботи, чого чекає від підприємства, яке буде потрібно навчання для ефективного виконання обов'язків, хто може охарактеризувати його як фахівця, з'ясувати можливість одержання рекомендацій.

### **Характеристика останнього місця роботи**

Бажано одержати інформацію про: підприємство, у якому працює або працював кандидат, сферу його діяльності, обсяги виконуваних робіт, місце і посаду даного

працівника в структурі підприємства, рівні відповідальності, умови організації трудової діяльності, рівень устаткування робочих місць. Також слід прояснити причини пошуку нової роботи, позитивні і негативні сторони попередньої діяльності кандидата, встановити період повідомлення про припинення трудових відносин, можливий час початку роботи у підприємстві.

### **Одержувана винагорода**

Для того, щоб у наступному більш точно визначити оплату роботи для кандидата, необхідно точно з'ясувати систему оплати на попередньому місці роботи, додаткові умови і пільги, які мав кандидат, розмір оплати за місяць і його чекання, пов'язані з матеріальною винагородою на новому місці роботи, рівень добробуту родини.

### **Індивідуальні характеристики**

Оскільки підприємство наймає не тільки фахівця, але й сподівається одержати лояльну до підприємства, морально стійку людину, необхідно прояснити його сімейний стан, цивільний статус, його захоплення. Особливу увагу треба приділити поведінковим реакціям кандидата під час інтерв'ю, спробувати сформулювати для себе припущення про його звичайну поведінку в робочій обстановці, інтелектуальні здібності, особистісні якості і властивості характеру, моральні характеристики, спробувати з'ясувати, як він "впишеться" у діючий колектив, рівень конфліктності, працездатності. [4]

Так як підбір персоналу є досить багатокроковим і витратним процесом, то гарною ідеєю буде автоматизувати або хоча б скоротити використання інформаційних технологій певні кроки в ньому. Одним з можливих вирішень задачі скорочення використання людських ресурсів для підбору персоналу є використання машинного навчання та нейромереж як його частини.



### ***2.1.2 Використання алгоритмів машинного навчання***

Для оптимізації вище зазначених процесів машинне навчання буде корисне з декількох причин. Перш за все інтелектуальна система, котра може самостійно встановлювати асоціації між вхідними даними, які надходять з анкети чи резюме кандидата просто прибирають один з кроків процесу рекрутингу відсіювання та оцінку пріоритету того який кандидат буде найкраще виконувати покладені на нього обов'язки базуючись на тому, як з цим справлялися люди до нього і якими характеристиками вони володіли. Іншою причиною є те, що система постійно покращується в процесі роботи і при цьому в ній немає певних людських слабкостей – вона не звільниться з роботи і не захворіє. З іншої ж сторони така система має виступати лише у вигляді певного «консультанта» залишаючи право вибору за людиною.

Однак в такій системі є і недоліки. Перший недолік кількість ресурсів, які використовуватимуться на підтримку мережі, поки вона навчається і не ще не дає точних прогнозів.

Ще в недалекому минулому потужностей обчислювальних машин не вистачало для обробки великих датасетів (datasets), які б ґрунтувалися на даних з зовнішнього світу (зображення, відео, величезні масиви даних на зразок статистики про працевлаштування ), або ж це займало неприйнятно багато часу і ресурсів. Використання глибоких мереж (deepe networks) на мобільних чи не дуже потужних пристроях, на яких ресурсів не так багато, можна забезпечити за допомогою методів стиснення глибоких моделей (deep model compression methods), що надасть можливість використовувати їх набагато менше.

Видів методів стиснення (deep model compression methods) за останні роки з'явилося досить багато. Приблизною класифікацією цих методів можна вважати наданий нижче список видів подібних методів

– методи обміну параметрами (parameter sharing methods)

- методи обрізання мережі (network pruning methods)
- «темні знання» ('dark knowledge')
- методи навчання учителя-ученика (teacher-student methods)
- методи декомпозиції матриць (matrix decomposition methods)

Ці методи, в основному, зосереджені на зменшенні складності глибоких моделей. Після стиснення одним із цих методів модель все одно доведеться «розпаковувати».

Одним з маловивчених методів, котрий може вирішити цю проблему, є метод навчання учителя-студента (teacher-student method) [4] для стиснення глибоких моделей. Це достатньо простий підхід, де неглибока мережа (студент) навчається у глибокої мережі (вчителя).

Проблема неглибокої моделі у низькій точності в обмін на економію ресурсів та зменшення складності обробки. За таких умов досягти точності, як у глибокої моделі-вчителя, потребує набагато більше часу, що не є добре. Тоді, як глибока мережа досить швидко але з використання більшої ресурсів досягає високої точності. Описані нижче методи використовуються для підвищення точності мережі-ученика без збільшення її глибини, а відповідно і кількості ресурсів.

Методи стиснення моделей були створенні відносно недавно і їх можна розділити на 4 типи що будуть описані нижче.

**Метод обміну параметрами** (parameter sharing methods) передбачає використання простої (не затратної) [5] хеш-функції для групування ваг (параметрів) в певні хеш-групи (hash buckets) для розділення параметрів, де кожна хеш-група відповідає одному параметру. В методі використовується k-вимірна кластеризація [7] для повного квантування (розбивка діапазону їх значень на скінчену кількість інтервалів) параметрів пов'язаних між собою шарів моделі. Цей метод дозволяє, судячи з проведених експериментів, підвищити ступінь стиснення моделі в десятки

разів (в одному з дослідів модель була стиснута в 24 рази), при втраті лише 1% точності. В даному методі регуляризація використовується замість прямого квантування параметрів лише зв'язаних шарів.

**Обрізання мережі** (network pruning methods) - полягає у відкиданні параметрів, вага яких нижче певного порогового значення заданого до навчання мережі. Цей метод можна розширити з використанням кодування Гаффмана для ще більшого скорочення кількості параметрів. Метод, в основному, спрямований на скорочення кількості обчислень та ігнорування фільтрів, які мають найменший вплив на точність. Також в рамках цього методу мережу можна дослідити надлишкові нейрони.

**«Темні знання»**(«dark knowledge»). Ключовою ідеєю цієї групи методів (до яких входить метод навчання учителя-студента (teacher-student method)) в навчанні мережею, яка має більшу глибину та складність та прості моделі котрі виконують схожі задачі. Метод навчання учителя-студента (teacher-student method) був запропонований С. Vucilu, котрий помічав штучні(тестові) дані без вказівки на вчителя, від якого вони були надані. Ці мітки потім використовувалися в тренуванні меншої моделі (студента). Однією з реалізацій цього методу передбачається імітація логіт змінних (logit values) моделі вчителя. Також в цьому методі можна використовувати проміжні результати прихованих шарів моделі як цільові значення для моделі студента. Потім цей метод було узагальнено введенням певної температурної змінної в softmax функцію. Це призвело до пом'якшення результуючих даних та збільшення значення змінної температури котра надає досить важливу інформація. Ця інформація, яка виражається у відносній оцінці (в ймовірності) вихідних класів, називається «темні знання».

**Метод декомпозиції матриць** (matrix decomposition methods) [6] полягає у використанні низькорівневої декомпозиції для стиснення ваг в різних шарах мережі. Метод перетворює щільні матриці ваг у повністю з'єднані шари до вигляду декомпозиції Тензора (Tensor decomposition), що значно зменшує кількість вхідних параметрів.

Стиснення глибоких моделей (compression of deep models) зарекомендувало себе у вирішенні трьох проблем: використання пам'яті, зменшення часу, витраченого на тренування моделі, та зменшення складності виконання. Як методи обміну параметрами (parameter sharing methods) так і метод декомпозиції матриць (matrix decomposition methods), зосередженні лише на зменшенні використання пам'яті глибокими моделями, але ці методи не зменшують час виконання (або час тренування), чи складність. Метод вчителя-студента (teacher-student method), навпаки, зосереджений на зменшенні складності та часу.

В рамках методу вчителя-студента (teacher-student method) в глибокому навчанні (deep learning), вчитель - це попередньо підготовлена та навчена глибока модель, яка використовується для навчання іншої (зазвичай неглибокої) моделі, яку називають учеником. Подібний метод має як плюси, так і обмеження, які були описані вище в розділі «Методи стиснення моделі». У використанні методу вчителя-студента є такі переваги:

- «темні знання», які присутні в результатах роботи моделі-вчителя, працюють як певні регуляризатори для моделі студента, так як вони забезпечують більш «м'який» набір знань, за яким легше відсіяти корисну інформацію;
- конвергенція (сходимість) зазвичай швидша ніж при використанні булевих міток завдяки м'яким цілям що пришвидшують тренування;
- відносно невелика кількість даних для тренування моделі-учня.

Наведені вище переваги дозволяють використати метод шумового (noise-based) регуляризатора для моделі вчителя. Далі буде описана система з одного вчителя та студента в системі вчитель-студент, яка є базою для наступних експериментів.

## 2.2 Математична модель оцінки придатності кандидата

### 2.2.1 Задача оцінки придатності кандидата

Призначенням цієї задачі є визначення ступеню придатності кандидата відповідній посаді, за декількома критеріями.

Дано:

- Анкета з вхідними даними кандидата, з кількістю полів  $n$ .
- Масив вакансій кількістю  $m$  зі значеннями важливих полів для кожної вакансії, в вигляді матриці (2.1)

$$\begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix}, \quad (2.1)$$

де  $a_{ij}$  наявність того чи іншого поля в вимогах вакансії.

Змінні:

- Кількість важливих полів в анкеті  $k$ .
- $x_j$  - важливість пункту анкети, де  $j = \overline{1, n}$

Ціль задачі:

Знайти коефіцієнти відповідності між кандидатом, що заповнив анкету, та вакансією з вказаними до неї вимогами.

Для вирішення задачі був розроблений евристичний алгоритм наступного вигляду:

Крок 1 ДЛЯ кожної вакансії зв'язаної з вибраним кандидатом:

Крок 1.1 ПРИЗНАЧИТИ кількість умов вакансії

Крок 1.2 ПРИЗНАЧИТИ кількість умов що задовольняє кандидат 0

Крок 1.3 ЯКЩО кількість умов вакансії  $> 0$

Крок 1.3.1 ДЛЯ кожної умови

Крок 1.3.2 ЯКЩО в анкеті кандидата поле з відповідною умовою при виконанні порівняння за допомогою заданої операції повертає true перейти до кроку 1.3.4

Крок 1.3.3 ІНАКШЕ перейти до кроку 1.3.5

Крок 1.3.4 ПРИЗНАЧИТИ вимогу як виконану кандидатом

Крок 1.3.5 ПРИЗНАЧИТИ вимогу як не виконану кандидатом

Крок 1.4 ЯКЩО вимога виконана кандидатом збільшити кількість умов що задовольняє кандидат на 1

Крок 1.5 ПРИЗНАЧИТИ кількість задоволених вимог в відсотковому співвідношенні

Крок 1.6 Закінчити виконання

### ***2.2.2 Задача класифікації даних за набором критеріїв***

Кожен об'єкт даних представляється як вектор (точка) в  $r$ -вимірному просторі (упорядкований набір  $r$  чисел). Кожна з цих точок належить тільки одному з двох класів. Питання полягає в тому, чи можна розділити точки гіперплощиною розмірності  $(r-1)$ . Це - типовий випадок лінійної роздільності. Шуканих гіперплощин може бути багато, тому вважають, що максимізація відстані між класами сприяє більш точній класифікації. Тобто, чи можна знайти таку гіперплощину, щоб відстань від неї до найближчої точки було максимальним. Це еквівалентно тому, що відстань між двома найближчими точками, що лежать по різні боки гіперплощини, максимально. Якщо така гіперплощина існує, вона називається оптимальною розділяючою гіперплощиною, а відповідний їй лінійний класифікатор називається оптимально розділюючим класифікатором.

Дано:

- $X$  — множина описів об'єктів моделей.
- $Y$  — множина номерів класів.

Існує невизначена *цільова залежність*- відображення  $y^*: X \rightarrow Y$ , значення якої відомі лише на елементах кінцевої навчальної вибірки  $X^m = \{(x_1, y_1), \dots, (x_m, y_m)\}$ . Для вирішення задачі побудуємо алгоритм  $a: X \rightarrow Y$ , здатний класифікувати довільний об'єкт  $x \in X$ .

### 2.3 Огляд методів розв'язання задачі класифікації

**Задача класифікації** —задача, яка оброблює множину об'єктів (ситуацій), котрі можна розділити певним чином на класи. Існує кінцева множина об'єктів, про які відомо, яким класам вони належать. Подібна множина називається вибіркою. Клас до якого належать інші об'єкти є невідомим. Алгоритм вирішення цієї задачі повинен класифікувати довільний об'єкт з наданої на вхід алгоритму множини.

**Класифікувати об'єкт** — означає, призначити в вигляді номеру або текстової назви клас, до якого відноситься об'єкт.

**Класифікація об'єкта** — текстова або числова назва класу, що призначається алгоритмом класифікації після отримання результатів його застосування до даного об'єкту.

Математична статистика задачу класифікації визначає як задачу дискретного аналізу. При використанні машинного навчання задача класифікації вирішується з використанням штучних нейронних мереж чи одного зі схожих за цільовим використанням методів при постановці експерименту у вигляді навчання з учителем.

### Опис методів розв'язання

Основною математичною задачею, яку розв'язує даний комплекс, є задача класифікації. Задача класифікації полягає у тому щоб за певним набором параметрів визначити до якого класу належить.

Для більшої достовірності експериментів та порівняння будемо використовувати декілька алгоритмів та підходів машинного навчання. Першим з них буде описаний нижче - метод опорних векторів.

### 2.3.1 Метод опорних векторів

В нас є тренувальна вибірка даних, котра складається з  $n$  точок вигляду (2.2)

$$(\vec{x}_1, y_1), \dots, (\vec{x}_n, y_n), \quad (2.2)$$

де  $y_i$  дорівнює або 1, або -1, і кожен вектор визначає клас, якому належить точка  $\vec{x}_i$ . Кожен  $\vec{x}_i$  -  $p$ -вимірний дійсний вектор. При використанні методу опорних векторів необхідно знайти «максимально розділову гіперплощину», котра розділяє групу точок  $\vec{x}_i$ , для яких  $y_i = 1$ , та групу точок, для яких  $y_i = -1$ , і визначається таким чином, щоб відстань між цією гіперплощиною та найближчою точкою  $\vec{x}_i$  з кожної з груп була максимальною.

Будь-яка гіперплощина може бути виражена та записана як множиною точок  $\vec{x}$ , які задовольняють (2.3)

$$\vec{\omega} * \vec{x} - b = 0, \quad (2.3)$$

де  $\vec{\omega}$  є вектором нормалі до цієї гіперплощини (може бути не нормалізованим). Параметр  $\frac{b}{\|\vec{\omega}\|}$  показує зсув гіперплощини вздовж вектора нормалі  $\vec{\omega}$  від початку координат. Та де  $b$  коефіцієнт зсуву гіперплощини.

### Жорстке розділення

Тоді як тренувальні дані задовольняють умову лінійної роздільності, у такому випадку маємо можливість вибрати дві паралельні гіперплощини, такі щоб відстань між класами даних була найбільшою. Частина, обмежена двома гіперплощинами, називають «розділенням». Ці можна описати рівняннями (2.4) та (2.5)

$$\vec{\omega} * \vec{x} - b = 1 \quad (2.4)$$

$$\vec{\omega} * \vec{x} - b = -1 \quad (2.5)$$

Відстань між цими двома гіперплощинами -  $\frac{2}{\|\vec{\omega}\|}$ , тож для максимізації відстані між ними нам треба мінімізувати  $\|\vec{\omega}\|$ . Оскільки нам не потрібні точки котрі



потрапляють до даного розділення, потрібно додати таке обмеження: для кожного наведеного  $i$ , або (2.6) або (2.7)

$$\vec{\omega} * \vec{x} - b \geq 1, \text{ якщо } y_i = 1, \quad (2.6)$$

$$\vec{\omega} * \vec{x} - b \leq -1, \text{ якщо } y_i = -1, \quad (2.7)$$

З наведеного вище обмеження видно що всі точки лежатимуть з потрібного нам боку розділення.

Це може бути переписано як (2.8)

$$y_i(\vec{\omega} * \vec{x}_i - b) \geq 1, \text{ для всіх } 1 \leq i \leq n. \quad (2.8)$$

Ми можемо зібрати це все разом, щоби отримати задачу оптимізації:

«Мінімізувати  $\|\vec{\omega}\|$  за умови  $y_i(\vec{\omega} * \vec{x}_i - b) \geq 1$ , для  $i = 1, \dots, n$ »

$\vec{\omega}$  та  $b$ , котрі розв'язують задачу, з класифікатором,  $\vec{x} \rightarrow \text{sgn}(\vec{\omega} * \vec{x} - b)$ .

Важливим та очевидним наслідком, вище зазначеного, геометричного опису є те, що максимально розділова гіперплощина визначена тими  $\vec{x}_i$ , котрі лежать на мінімальній відстані від цієї ж гіперплощини. Ці  $\vec{x}_i$  прийнято назвати опорними векторами.

### М'яке розділення

Якщо ж ми бажаємо розширити використання методу на ті випадки, при яких вхідні дані не є лінійно роздільними, ми маємо ввести завісну функцію втрат (англ. hinge loss function) (2.9),

$$\max(0, 1 - y_i(\vec{\omega} * \vec{x}_i - b)), \quad (2.9)$$

Якщо обмеження показані в (2.9) задовольняється, тоді ця функція є нульовою, тобто, означають що  $\vec{x}_i$  лежить із правильного боку розділення.

Дані з іншого боку розділення це результат визначення результату цієї функції пропорційний відстані від нашого розділення.

Тобто потрібно мінімізувати наведену нижче (2.10)

$$\left[ \frac{1}{n} \sum_{i=1}^n \max(0, 1 - y_i(\vec{\omega} * \vec{x}_i - b)) \right] + \lambda \|\vec{\omega}\|^2, \quad (2.10)$$

де параметр  $\lambda$  означає можливий «компроміс» між збільшенням розділення та виконання умови розташування  $\vec{x}_i$  з правильного боку розділення.

За значень  $\lambda$  достатньо малих моделей з м'яким розділенням (англ. soft-margin) поводитиметься однаково відповідно до моделі із жорстким розділенням (англ. hard-margin SVM), за виконання умови, що вхідні дані є лінійно класифікованими, при цьому модель навчатиметься життєздатним правилам класифікації, якщо ні.

Так як вище в SVM алгоритмі використовується підхід навчання з вчителем, то спочатку необхідно виконати алгоритм навчання. Алгоритм навчання матиме такий вигляд:

### **Алгоритм складається з таких кроків:**

Крок 1.  $\lambda$  ініціалізувати значеннями, що задовольняють обмеження

Крок 1.1. ДЛЯ кожного  $\lambda_j$  вибрати перший коефіцієнт для оптимізації

Крок 1.2. ДЛЯ кожного  $\lambda_i$  в залежності від  $\lambda_j$  вибрати другий коефіцієнт для оптимізації

Крок 1.3. ДЛЯ кожного  $\lambda_j$  та  $\lambda_i$  виконати оптимізацію, та знайти нове значення  $\omega$ :

Крок 1.4. ЯКЩО поточні значення  $\lambda_j$  та  $\lambda_i$  відрізняються від попередніх

Крок 1.4.1. ПЕРЕЙТИ до кроку 1.2

Крок 1.5 ІНАКШЕ перейти до кроку 2

Крок 2.  $\lambda$  ініціалізувати значеннями, що задовольняють обмеження, наприклад  
( $\forall n : \lambda_n := 0, \omega_0 := 0, C := 10$ )

Крок 2.1 Побудувати чергу з прикладів навчальної вибірки  $\lambda_j$  та  $\lambda_i$

Крок 2.2 ДЛЯ кожного елемента черги

Крок 2.2.1 ВИКОНАТИ Крок 3 для поточного елемента черги

Крок 2.2.2 ЯКЩО оброблено не всі приклади з черги переходимо до кроку 2.2.1

Крок 2.2.3 ЯКЩО навчальної вибірки  $\lambda_j$  та  $\lambda_i$  змінила параметри виконуємо

Крок 2.5

Крок 2.3 Побудувати чергу з прикладів навчальної вибірки  $\lambda_j$  та  $\lambda_i$  для яких  
значення коеф.  $0 < \lambda_n < C$

Крок 2.4 ДЛЯ кожного елемента черги

Крок 2.4.1 ДЛЯ кожного елемента черги виконати Крок 3

Крок 2.4.2 ЯКЩО оброблено не всі приклади з черги переходимо до кроку 2.4.1

Крок 2.4.3 ЯКЩО навчальна вибірка  $\lambda_j$  та  $\lambda_i$  змінила параметри виконуємо Крок

2.3 Побудувати чергу з прикладів навчальної вибірки  $\lambda_j$  та  $\lambda_i$  для яких значення  
коеф.  $0 < \lambda_n < C$

Крок 2.4. ІНАКШЕ перейти до кроку 2.1

Крок 2.5. закінчити роботу

**Крок обробки навчального прикладу  $\lambda_j$**

Крок 3 ДЛЯ точки  $\lambda_j$

Крок 3.1 зайти можливу похибку

$$E_j := \begin{cases} E'_j ; 0 < \lambda_j < C \\ u(x_j) - y_j ; (\lambda_j \leq 0) \vee (\lambda_j \geq C) \end{cases},$$

де  $u$  функція класифікації  $x_i$  та  $y_i$  пара з навчального прикладу

Крок 3.2 ЯКЩО не порушуються вище задані умови перейти на Крок 3.5

Крок 3.3 виконати крок 4

Крок 3.4 ЯКЩО навчальна вибірка  $\lambda_j$  та  $\lambda_i$  змінила параметри виконуємо Крок

3.4

Крок 3.5 повернути стан параметрів, закінчити роботу

Алгоритм пошуку другого елементу пари  $\lambda_i$  для оптимізації

Крок 4 Побудувати чергу з  $\lambda$  для яких виконується  $0 < \lambda_n < C$

Крок 4.1 ЯКЩО черга пуста то перейти до Кроку 4.8

Крок 4.2 Серед елементів черги знайти  $\lambda_i$  для якого  $\max |E_j - E'_i|$

Крок 4.3 Виконати оптимізацію для  $\lambda_j$  та  $\lambda_i$

Крок 4.4 ЯКЩО параметри змінилися перейти до Кроку 4.10

Крок 4.4.1 Перемішати елементи черги випадковим чином

Крок 4.5 Вибрати перший елемент черги

Крок 4.5.1 Виконати оптимізацію для  $\lambda_j$  та  $\lambda_i$

Крок 4.5.2 Виконати зсув черги

Крок 4.6 ЯКЩО параметри змінилися перейти до Кроку 4.10

Крок 4.7 ЯКЩО в черзі залишилися приклади перейти до кроку 4.5

Крок 4.8 Побудувати чергу з усіх  $\lambda$ , перемішати всі елементи черги випадковим чином

Крок 4.9 Виконати крок 4.5

Крок 4.10 повернути стан параметрів, закінчити роботу

Іншим підходом до машинного навчання, котрий буде використано, буде навчання нейронної мережі методом учителя-ученика (teacher-student method) з використанням підходу дистиляції знань. Також підхід буде покращено використанням гаусівського шуму (Gaussian noise) до знань мережі вчителя.

### 2.3.2 Використання підходу дистиляції знань

Існує метод [1] навчання моделі студента з логарифмічною ймовірністю по змінній  $z$ , яка називається логіт функцією (logits) і є результатом шару перед викликом функції м'якого максимуму (softmax). Мережа виконує навчання за допомогою регресії з використанням логіт функції (logits) з навчальними даними поданими у вигляді  $\{(x^{(1)}, z^{(1)}), \dots, (x^{(i)}, z^{(i)}), \dots, (x^{(n)}, z^{(n)})\}$ .

Функція втрат (L2) буде мати наступний вигляд (2.11):

$$L(x, z, \theta) = \frac{1}{2T} + \sum_i \|g(x^{(i)}; \theta) - z^{(i)}\|_2^2, \quad (2.11)$$

де:

- $T$  розмір піднабору даних для однієї ітерації (mini-batch),
- $x^{(i)}$  це  $i$ -тий навчальний рядок в піднаборі (mini-batch),
- $z^{(i)}$  це відповідний вивід логіт функції (logits) попередньо навченого вчителя для  $x^{(i)}$ ,
- $\theta$  – набір параметрів моделі студента,
- $g(x^{(i)}; \theta)$  вивід учнівської моделі логіт функції (logits) для  $x^{(i)}$ .

Далі дана ідея буде використовуватися для додання шумів (noise-based) в знання вчителя.

## 2.4 Модифікація методу вчителя-ученика

Навчання моделі студента з використанням логіт функції та гаусівських шумів

Продуктивність неглибоких моделей в рамках методу вчителя-студента (teacher-student method) була значно покращена за допомогою методів запропонованих Хінтоном (G. Hinton)) [9]. Але, як себе поведе модель, якщо її навчає декілька вчителів. Аналогічно реальному світові, де студент може покращити швидкість і якість навчання одного предмету, отримуючи знання про нього від декількох викладачів (з альтернативними думками, повторенням вже пройденого і т.д.), можна припустити, що схожим чином поведе себе і модель студента в рамках методу вчитель-ученик (teacher-student method). Але замість використання декількох вчителів (що збільшить кількість вхідних даних, а відповідно і час навчання) використовується методологія симуляції ефекту навчання в декількох вчителів шляхом введення «шумів» та «заплутувань» в початкові знання моделі вчителя. Результатом заплутувань є не лише імітація навчання від декількох вчителів, а також призводить до шумів в шарі втрат (loss layer), що створює ефект регуляризатора. Таким чином, цей новий зашумлений вчитель є регуляризатором і допомагає студентам краще навчатися і отримувати результати ближчі до того якими вони є у вчителя. Метод має назву «Логіт збурення» («Logit Perturbation»).

Нехай  $\epsilon$  вектор гаусівського шуму (Gaussian noise) з середнім  $\mu = 0$  і середнім відхиленням  $\sigma$ . Значення  $\epsilon$  дорівнює числу класів(логітів) у мережі вчителя. Якщо  $z^{(i)}$  результат вихідного шару моделі вчителя для  $x^{(i)}$ , то  $z^{(i)}$  змінюється як (2.12):

$$z^{(i)} = (\mathbf{1} + \epsilon), \quad (2.12)$$

де  $\mathbf{1}$  - одиничний вектор та  $i \in R^n$ ,  $n$  – кількість класів. Функція втрат буде виглядати наступним чином (2.13):

$$L(x, z, \theta) = \frac{1}{2T} \sum_i \|g(x^{(i)}; \theta) - z^{(i)}\|_2^2, \quad (2.13)$$

Значення  $\sigma$  визначає визначає збурення, тобто більше  $\sigma$  – більше збурення оригінальних значень логіт функції вчителя  $z^{(i)}$ . Але це збурення не потрібно використовувати для всіх наборів. Замість цього потрібно вибрати набори з певної міні-вибірки з деякою ймовірністю  $\alpha$ , а тоді значення логіту обраних наборів збурюються за допомогою рівняння (2). Де  $\theta$  ваги нейромрежі. Отримавши модель студента з початковими вагами  $\theta_0$ , ми знайдемо кінцеві параметри  $\theta$ , використовуючи метод стохастичного градієнта (Stochastic gradient descent), де в  $(t + 1)$  ітерації - параметр  $\theta$  змінюється наступним чином (2.14):

$$\theta_{(t+1)} = \theta_t - \gamma_t \cdot \sum_{(x,y) \in D_t} \nabla_t [L(x, z, \theta)], \quad (2.14)$$

Тут  $T = |D_t|$  - розмір міні-вибірки, взятий випадково з навчальної вибірки  $D$ ,  $\gamma_t$  - швидкість навчання,  $L(x, z, \theta)$  - рівняння (3),  $\nabla_t [L(x, z, \theta)]$  обчислюється з використанням методу зворотного поширення помилки (gradient backpropagation).

Таким чином, деякі набори відбираються з ймовірністю  $\alpha$  з міні-вибірки. Цільовими значеннями логіт функції збурюються за допомогою рівняння (2). Функція втрат студентської мережі визначається рівнянням (3).

Доведено, що «зашумлені» дані допомагають регуляризувати модель. додавання регуляризації в  $L$  функцію втрат еквівалентно додаванню гаусівського шуму (Gaussian noise) у вхідні дані. Регуляризована функція втрат виглядає як (2.15):

$$L(x, \theta, z) = L(x, \theta, z) + R(\theta), \quad (2.15)$$

де  $x$  є гаусівським шумом (Gaussian noise),  $L$  еквівалентна шумним вхідним даним  $L(x, \theta, z)$ , а  $R(\theta)$  – регуляризатор  $L$  [3]. У вибраному методі ми збурюємо цільові вихідні данні  $z$  (шумні дані), замість вхідних даних  $x$ . Тепер покажемо, що збурення цільових вихідних даних  $z$ , значення логіт функції вчителя, еквівалентно додавання

шумного (noise-based) регуляризатора, до функції втрат. З рівняння (2.12) отримуємо (3.16):

$$z^{(i)} = (1 + \varepsilon) * z^{(i)} = z^{(i)} + \varepsilon z^{(i)}, \quad (2.16)$$

Отже, можемо переписати рівняння втрат  $L$  (2.13): як (2.17):

$$\begin{aligned} L(x, \theta, z) &= \|z^{(i)} - g(x^{(i)}, \theta) - \varepsilon \cdot z^{(i)}\|_2^2 \\ &= \|z^{(i)} - g(x^{(i)}, \theta)\|_2^2 + \|\varepsilon \cdot z^{(i)}\|_2^2 \\ &= \|z^{(i)} - g(x^{(i)}, \theta)\|_2 * \|\varepsilon \cdot z^{(i)}\|_2 = L(x, \theta, z) + E_R, \end{aligned} \quad (2.17)$$

де  $E_R = \|\varepsilon \cdot z^{(i)}\|_2^2 + 2\|z^{(i)} - g(x^{(i)}, \theta)\|_2 * \|\varepsilon \cdot z^{(i)}\|_2$  - новий регуляризатор, який виведений на основі шуму  $\varepsilon$ . Таким чином, збурення логітів у мережі вчителя еквівалентно додаванню зашумленого (noise-based) регуляризатора до функції втрат.

## 2.5 Результати досліджень ефективності методу

Оцінка вибраного методу була проведена на декількох наборах даних, зокрема на MNIST [9]. Метод стохастичного градієнта (Stochastic gradient descent (SGD)) використовується для навчання всіх мереж, з розміром піднабору даних для однієї ітерації (mini-batch) рівному 64. Конвергенція (сходимість) була досягнута шляхом тестування на наборі підтверджених даних. В цьому розділі результати експериментів порівнюються з результатами і продуктивністю метода [1]. Результати на всіх наборах даних виглядають досить перспективно, найкращого результату вдалося досягти на наборі даних CIFAR-10.

### Використання MNIST

MNIST [9] – популярний датасет для тренування моделей розпізнавання рукописного вводу з 10 класами (0-9). Навчальний набір містить 50000 зображень та підтверджених з них 10000. Всі зображення в розмірі  $28 \times 28$  та кольорами які є градаціями сірого кольору. Тренувальні данні попередньо не обробляються, дозволяючи виконувати власну попередню обробку на цих наборах даних.



*Модель вчителя.* У якості мережі вчителя використана модифікована мережа LeNet, котра має 2 згорткові шари і повнозв'язаний шар з 10-ма класифікаторами. (кофігурація виглядатиме так [C5 (S1P0) @ 20-MP2 (S2)])

*Мережа студент.* У якості мережі студента використана невелика мережа з двома повнозв'язними шарами з 800 нейронами в кожному шарі. Архітектура може бути закодована як: FC800-FC800-FC10

*Результати.* Модель вчитель отримала 68 помилок в тестових наборах (з 10000 тестових зразків, частота спостереження помилки = 0,0068). Студентська мережа допустила 97 помилок (частота спостереження помилки = 0,0097) при базовому методі вчитель-студент (метод логіт-регресії). Так як різниця в продуктивності між вчительською та студентською мережею невисока то ймовірність відбору вибірки  $\alpha = 0,15$ , тобто приблизно 15% піднаборів кожного набору параметрів відбираються для збурення. Збурення відбувалося при різних рівнях гаусівського шуму ( $\mu = 0$ , різними  $\sigma$ s), як показано в таблиці 1. Цей шум доданий безпосередньо до ненормалізованих логітів у всіх наших експериментах у цьому дослідженні. Ми бачимо, що відбувається послідовне покращення роботи студента при застосуванні обурення до логіт.

Як видно з таблиці 2.1, використання гаусівського шуму для збурення ненормалізованих логітів, зменшує кількість помилок, а значить поліпшує роботу мережі студента.

Таблиця 2.1 Результати експериментів на MNIST

Рівень шуму ( $\sigma=\text{std}$ )	Частота спостереження помилки	Покращення %
0.10	0.0096	1.0%
0.20	0.0093	4.1%
0.30	0.0094	3.1%
0.40	0.0087	10.3%
0.50	0.0087	10.3%
0.60	0.0090	7.2%
0.70	0.0090	7.2%

Рівень шуму ( $\sigma=\text{std}$ )	Частота спостереження помилки	Покращення %
0.80	0.0086	11.3%
0.90	0.0086	11.3%
1.00	0.0087	10.3%

В другому експерименті був використаний датасет SVHN[10] – величезним датасетом реальних знімків знаків з номерами будинків з зображеннями в тому числі з Google Street View. Цей датасет дещо менший ніж MNIST. В ньому більше 70000 навчальних зображень, 26000 зображень для тестів. Кожне зображення має розмір 32x32.

*Модель вчителя.* Як вчитель використана мережа Network-in-Network з конфігурацією [C5(S1P2)@192]-[C1(S1P0)@160]-[C1(S1P0)@96-MP3(S2)]-D0.5-

[C5(S1P2)@192]-[C1(S1P0)@192]-[C1(S1P0)@192-AP3(S2)]-D0.5-[C3(S1P1)@192]- [C1(S1P0)@192]-[C1(S1P0)@10]-AP8(S1)

*Мережа студент.* Студентом буде немодифікована LeNet з конфігурацією [C5(S1P2)@32-MP3(S2)]- [C5(S1P2)@64-MP3(S2)]- FC1024-FC10. В мережі буде два шари для того щоб відповідати умові глибини мережі студента відносно мережі учителя.

Модель учитель отримала на виході 3.82% помилок на тестовому наборі SVHN. Модель учня при немодифікованому методі учителя-учня допустила 4.6% помилок на тому самому датасеті. Так як і в минулий раз різниця між вчителем та учнем не є досить значною тому як і в попередньому випадку було обрано  $\alpha = 0,15$ . Після використання модифікованого методу навчання нейронної мережі модель ученик досягла частоти спостереження помилки 4.45% в межах розглянутого діапазону стандартних відхилень в гаусівському шумі. Детальніше експеримент наведено нижче в таблиці 2.2. Також можна зробити висновок що завеликий рівень шуму

призводить до погіршення результатів котрі видає модель на цьому датасеті, що в свою чергу дає зрозуміти наскільки важливо правильно підбирати параметри  $\alpha$  та  $\sigma$ .

Таблиця 2.2 Результати експериментів на SVHN

Рівень шуму ( $\sigma=\text{std}$ )	Частота спостереження помилки	Покращення %
0.10	4.51	2.0%
0.20	4.46	3.0%
0.30	4.45	3.3%
0.40	4.54	1.3%
0.50	4.57	0.7%
0.60	4.51	2.0%
0.70	4.61	-0.2%
0.80	4.68	-1.7%
0.90	4.72	-2.6%
1.00	4.81	-4.6%

## Висновок

З усього вище написаного можна зробити декілька висновків. Перш за все використання алгоритмів машинного навчання має декілька важливих недоліків, як наприклад сама необхідність перше виконувати навчання моделі для того аби вона могла встановлювати асоціативні зв'язки, що витрачає досить багато ресурсів та часу. Цей недолік можна згладити або повністю прибрати за допомогою використання нейронних мереж та підходу навчання teacher-student завдяки по суті прибирання кроку першого навчання нейромережі. Натомість виникає інша проблема. Проблема навчальної вибірки яка може бути досить обмеженою. Для вирішення ж цієї проблеми і був розроблений підхід збурення «знань» за допомогою гаусівського шуму. Такий підхід допомагає імітувати навчання від декількох вчителів, що в свою чергу збільшує точність нейромережі.

### 3 ОПИС ПРОГРАМНОГО ТА ТЕХНІЧНОГО ЗАБЕЗПЕЧЕННЯ

#### 3.1 Засоби розробки

Для створення програмного продукту (системи) було використано такі засоби як мова програмування PHP та фреймворк Phalcon On PHP, бази даних MySQL та Redis

Мова PHP є високорівневою мовою, яка дуже проста у використанні, з великою кількістю бібліотек.

Мова PHP - інтерпретована мова, котра не є компільованою а є інтерпритовною. Завдяки цьому не використовується час на компіляцію тому і зміни в програми можна вносити оперативно та швидко. Також важливою ознакою є те, що код пишеться максимально швидко, та помилки гарно відловлюються стандартними засобами.

Фреймворк Phalcon On PHP хоча є досить популярним до найвикористовуваних не належить. Він написаний на мові C як розширення до мови PHP. Він застосовується для створення веб-застосунків. Це ідеальний фреймворк для найбільш швидкого написання робочих прототипів, адже він має велику кількість готових та простих у використанні бібліотек.

Система управління базами даних MySQL - об'єктно-реляційна система керування базами даних.

На відміну від більшості інших СКБД MySQL не контролюється якоюсь однією компанією, її розробка можлива завдяки співпраці багатьох людей та компаній, які хочуть використовувати цю СКБД та впроваджувати у неї найновіші досягнення. Великою перевагою цієї СКБД є безліч вбудованих типів даних, її швидкість, та простота у розширенні. Також вона має високу надійність.

Сховище Redis – нереляційна база даних типу Ключ-Значення. Так як система планується як високонавантажений сервіс буде використовуватися як система кешування та зберігання простих і динамічних структур даних .

Мова HTML (англ. HyperText Markup Language — Мова розмітки гіпертекстових документів) — стандартна мова розмітки веб-сторінок в Інтернеті.

Більшість веб-сторінок створюються за допомогою мови HTML (або XHTML). Документ HTML оброблюється браузером та відтворюється на екрані у звичному для людини вигляді.

Каскадні таблиці стилів (англ. Cascading Style Sheets або скорочено CSS) — спеціальна мова, що використовується для відображення сторінок, написаних мовами розмітки даних.

Мова JavaScript — назва реалізації стандарту мови програмування ECMAScript компанії Netscape, базується на принципах прототипного програмування. Найпоширеніше і найвідоміше застосування мови — написання сценаріїв для веб-сторінок.

## **3.2 Архітектура програмного забезпечення**

### **3.2.1 Загальні положення**

Фрон-енд частина реалізована за допомогою мови програмування JavaScript а саме технології ReactJS. Що дозволяє без перезавантаження сторінки змінювати елементи прямо в браузері.

Фреймворк ReactJS- це відкрита JavaScript бібліотека для створення інтерфейсів користувача, яка покликана вирішувати проблеми часткового оновлення вмісту веб-сторінки, з якими стикаються в розробці односторінкових застосунків. React обробляє тільки користувацький інтерфейс . Це відповідає видові у шаблоні MVC,

При розробці бек-енд частини даного програмного забезпечення було використано архітектурний шаблон MVC. Модель–вигляд–контролер — архітектурний шаблон, який використовується під час проектування та розробки програмного забезпечення.

Перевага використання шаблону MVC — чітке розділення логіки представлення інтерфейсу користувача, а також логіку застосунку. Також дуже зменшує складність великих застосунків, стиль написання коду в результаті є

набагато структурованими, чим полегшує тестування застосунку а також повторного використання певних рішень.

Архітектурний шаблон MVC передбачає поділ системи на три взаємопов'язані частини: модель даних, вигляд (інтерфейс користувача) та модуль керування. Застосовується для відокремлення даних (моделі) від інтерфейсу користувача (вигляду) так, щоб зміни інтерфейсу користувача мінімально впливали на роботу з даними, а зміни в моделі даних могли здійснюватися без змін інтерфейсу користувача.

Блок Модель (Model) — це дані та правила, що використовуються для роботи з даними, які представляють концепцію керування застосунком. Вона має містити в собі як дані, так і логіку предметної області. Це зазвичай інформаційна, логічна модель, яка може бути також відображенням об'єкту реального світу. Модель дає контролеру представлення даних, які запросив користувач. Тобто, модель реалізує всю логіку роботи програми і підготовлює дані для відображення (View).

Блок відображення (View) — забезпечує різні способи представлення даних, що є отриманими з моделі. Тобто, представлення візуалізує результати роботи програми.

Блок Контролер (Controller) — керує запитами користувача, що отримуються в вигляді HTTP запитів, які здійснюються при взаємодії користувача з відображенням, при натиску на елементи інтерфейсу. Контролер забирає дані з моделі на стороні сервера і передає їх по мережі з однієї сторони, а з іншої на стороні клієнта контролер ловить дані і передає їх для візуалізації. В таблиці 3.1 описано які діаграми буде доцільно побудувати в рамках дисертації.

Таблиця 3.1 - Доцільна побудова наступних діаграм UML фази уточнення

№	Тип діаграми	Роль у процесі проектування
1	Діаграма послідовності	У першому наближенні визначає набір класів та їхню послідовність взаємодії для вирішення певної задачі

№	Тип діаграми	Роль у процесі проектування
2	Діаграма класів	Визначає набір класів та їхню взаємодію задля вирішення певної задачі (виконання процесу)
3	Діаграма компонентів	Визначає групування класів у компоненти (а затим у виконувани або бібліотечні двійкові файли) та ланцюжки виклику компонентів іншими компонентами під час виконання програми.

### Діаграма послідовності

Діаграма послідовності, зображена на рисунку 3.1, для створення нової вакансії для надання можливості кандидатам подавати анкети на відкриту вакансію.

На діаграмі задіяні наступні класи описані в таблиці 3.2:

Таблиця 3.2 – Класи діаграми послідовності

Клас	Відповідальність
Recruter	Створення вакансії, встановлення вимог до вакансії, відбір кандидатів
VacanciesContoller	Клас контролює відображення вакансії, її вивід користувачу, і логіку відображення інформації за'язаної з вакансіями
VacanciesModel	Клас контролює поведінку вакансії, її активність, і дає можливість змінювати ряд параметрів вакансії
InputsModel	Дає можливість визначати важливі пункти анкети, контролює ступінь важливості цих пунктів



Клас	Відповідальність
WeighthsModel	Умови вакансії, яким повинні задовольняти Inputs, дозволяє змінювати, додавати чи корегувати їх

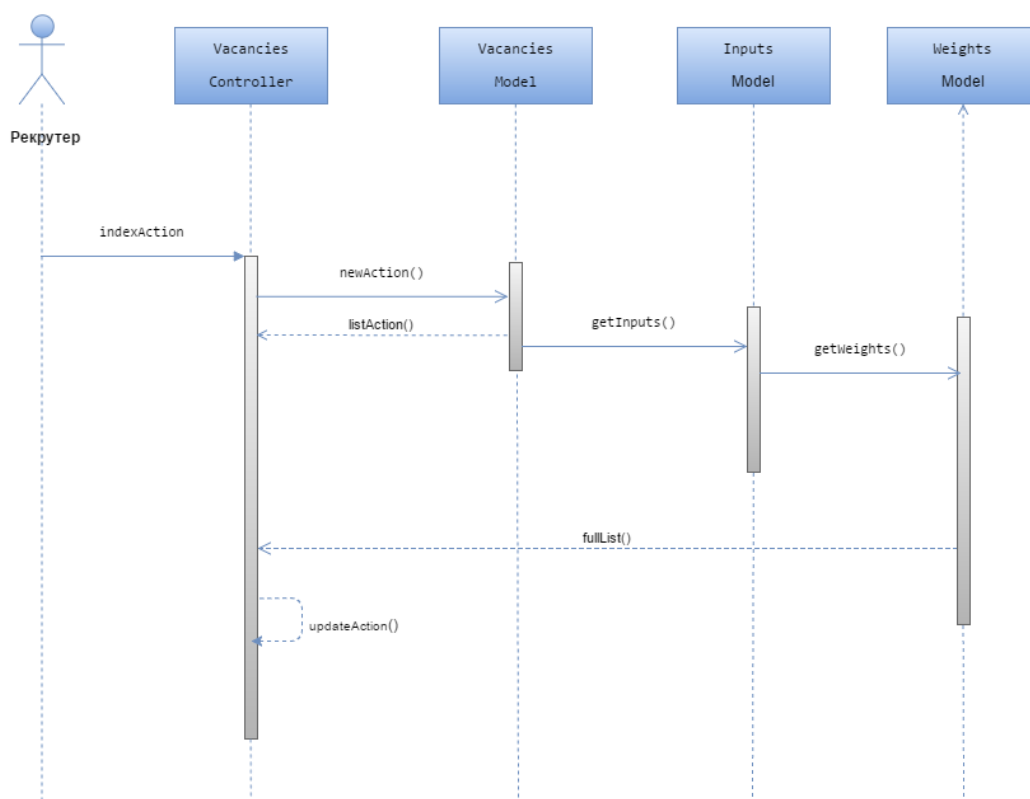


Рисунок 3.1. - Діаграма послідовності для прецеденту *Встановлення вимог до вакансії* в системі підбору кандидатів для HR менеджменту

### Діаграма класів

На рисунку 3.2 представлена структурна схема класів, які відповідають за виконання таких функцій програми, як встановлення з'єднання з СУБД MySQL, робота з даними, робота з транзакціями, складання та прогнозування плану випуску продукції та формування звітів «План виробництва».

Діаграма містить п'ять класів, а саме:

- «Vacancy» - клас який відповідає за створення, редагування і т.д.;

- «Inputs» - клас, що відповідає за умови які задаються для вакансії, і під які має підпадати кандидат;
- «Weights» - клас у якому керує пунктами анкети кандидата;
- «Candidates» - клас що відповідає за керування кандидатами, а також підбирає найкращі вакансії для нього.

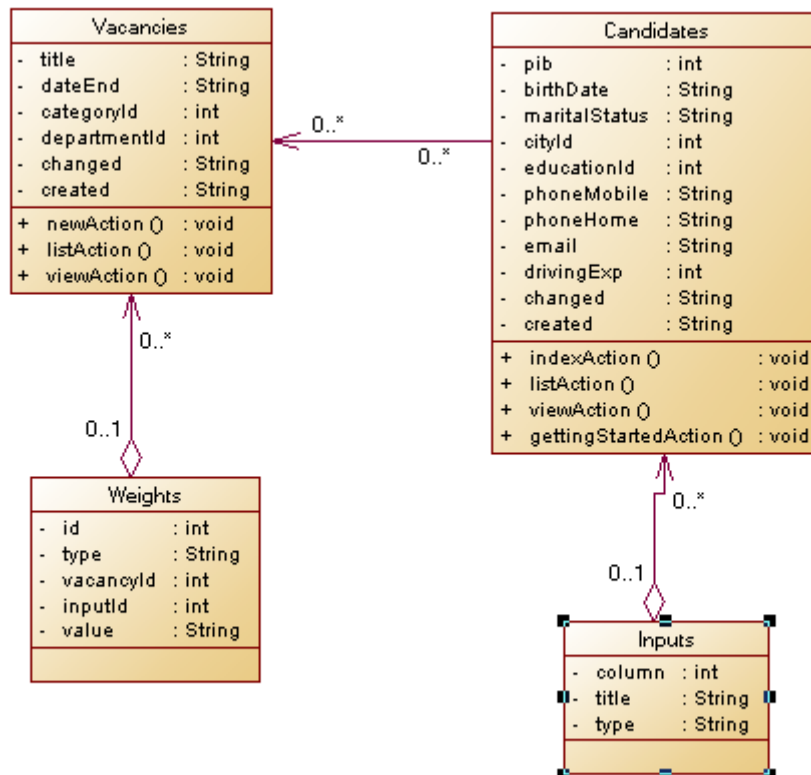


Рисунок 3.2 – Структурна схема класів

### Діаграма компонентів

Система підбору кандидатів для HR менеджменту функціонує на стеку LEMP (Linux, Nginx, MySql, PHP), система є веб застосунком і працює таким чином:

- Так як це веб застосунок, то перш за все певний запит вводить в адресний рядок клієнтського застосунку (браузеру);
- Потім сервер (Nginx) звертається до точки входу застосунку – файлу index.php;

- Файл вииконується інтерпретатором мови PHP. А сам файл перенаправляє його на інші аналогічні в яких реалізується логіка програми;
- Зв'язок з БД реалізується через розширення мови php для роботи з СУБД MySQL – php-mysql;
- Самі ж дані знаходяться в релятивістській БД , яка керується мовою SQL;
- Phalcon on PHP – високопродуктивний фреймворк мови php написаний на C, і є по суті розширенням мови PHP.

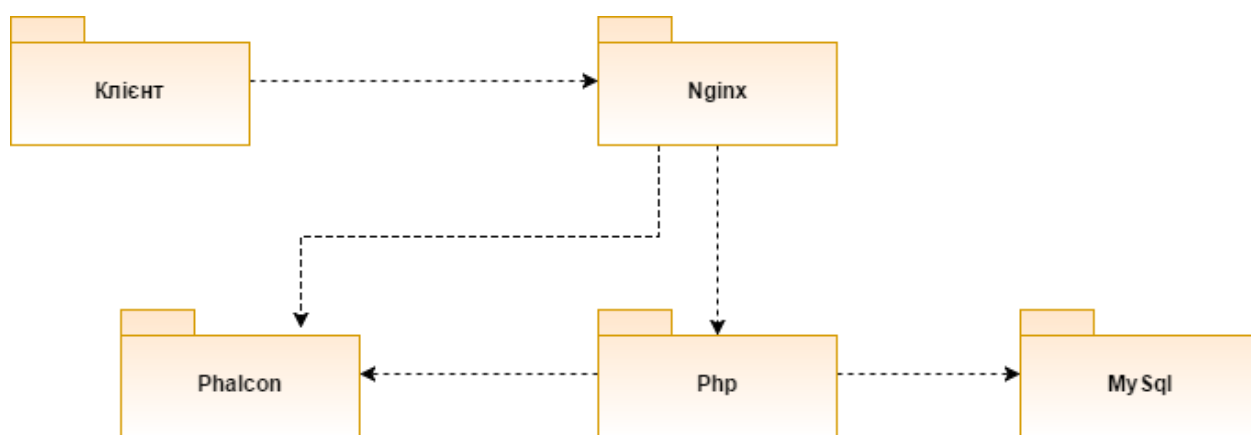



Рисунок 3.3. - Діаграма компонентів застосування

### 3.3 Настанова користувача

Для запуску програмного застосування потрібно ввести в адресний рядок браузера адресу сайту, на момент розробки адреса recruit.dev

Першою сторінкою при вході буде сторінка авторизації показана на рисунку 3.4



Рекрутингова система

Введіть email

Введіть пароль

Ввійти на сайт

Рисунок 3.4 – Сторінка авторизації

Після авторизації ви опинитесь на сторінці заповнення анкети кандидата, де потрібно заповнити всі поля анкети та натисну на кнопку «Створити кандидата» (рисунок 3.5)

Recruiting

Новий кандидат

Кандидати

Вакансії

Статистика

Про систему

Юрій

Recruitment Project

Source code on Github

Створення нового кандидата

Основна інформація

Категорія посади

Вільнонаймана

ПІБ

Введіть, будь ласка, ПІБ повністю

Сімейний стан

Неодружений(а)

Контактна інформація

Адреса проживання

Мобільний телефон

+38(099)012-3123

Домашній телефон

+38(035)492-1575

Освіта

Освіта

Вища

Профіль освіти

Право

Місце навчання

Навики

Власний транспорт

Посвідчення водія (номер)

Робота на ПК

Рекомендації

Рекомендації

Зберегти

© Recrutial Team 2017

Рисунок 3.5 – заповнення заявки

Будучи авторизованим є можливість використовувати меню сайту, показане на рисунку 3.6 (зображення інверсовано для покращення сприйняття). Є можливість перейти до пункту меню керування вакансіями, керування кандидатами, та пункту «Про систему»



Рисунок 3.6 – меню сайту

Якщо вибрати в меню пункт, кандидати, то можна буде переглянути всіх кандидатів на сторінці кандидатів.(рисунок 3.7)

**Список кандидатів**

■ - відібрані кандидати  
■ - кандидати, що проходять відбір  
■ - не прийняті кандидати  
 # - номер кандидата

#	ПІБ	Адреса	Водіння	Робота за ПК	Створено	Змінено	Дія
1	asdasd		- ()		2017-05-10 04:54:42	2017-05-10 04:54:42	t
2	Талько Юрій Сергійович		машина ()		2017-05-12 06:51:28	2017-05-12 06:51:28	t
3	Талько Юрій Сергійович1	м. Житомир, Вул котовського 976	машина ()	норм	2017-05-12 11:11:11	2017-05-12 11:11:11	t
4	Юрів Юрій Сергійович	м. Київ, Вул котовського 976	машина (123123)	фівфів			t
5	Юрів Юрій Юрієвич	м. Київ, Вул котовського 976	машина (фів 123123)	норм			t

1/1

[Перша](#)
[Попередня](#)
[Наступна](#)
[Остання](#)

© Recrutal Team 2017

Рисунок 3.7 – сторінка всіх кандидатів

Для того щоб переглянути більш докладну інформацію про кандидата потрібно натиснути на його ім'я. Тоді відкриється нова сторінка з докладною інформацією про кандидата (рисунки 3.8-3.9) взятою з анкети кандидата, а також його відповідність наявним вакансіям і відсотках (рисунок 3.10).

## Кандидатура Юрив Юрій Сергійович

🕒 Останнє оновлення:



	Юрив Юрій Сергійович
	+38(012) 312-3123
	+38(012) 312-3123
	
<b>Рекомендації</b> фівфів	
<b>Адреса</b> м. Київ, Вул котовського 97б	
<b>Досвід роботи з ПК</b> фівфів	
<b>Навчання</b> КПІ	
<b>Номер посвідчення водія</b> 123123	
<b>Prior Experiences</b>	
<b>Software Engineer at Twitter</b> Lorem ipsum dolor sit amet, ea vel prima adhuc, scripta liberavisse ea quo, te vel vidit mollis complectitur. Quis verear mel ne. Munere vituperata vis cu, te pri duis timeam scaevola, nam postea diceret ne. Cum ex quod aliquip mediocritatem, mei habemus persecuti mediocritatem ei.	
<b>Software Engineer at LinkedIn</b>	

Рисунок 3.8 – докладна інформація про кандидата частина 1

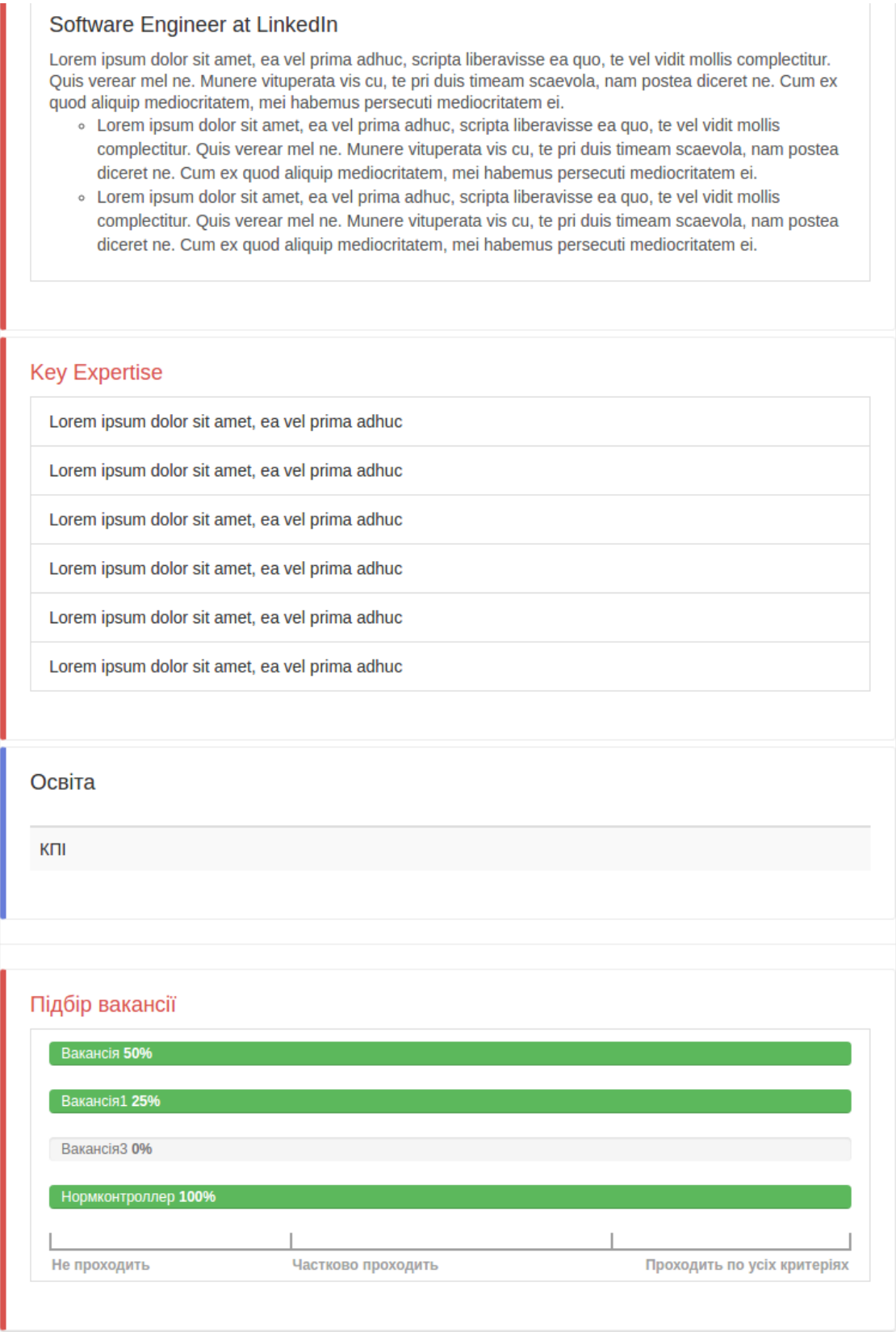


Рисунок 3.9 – докладна інформація про кандидата частина 2



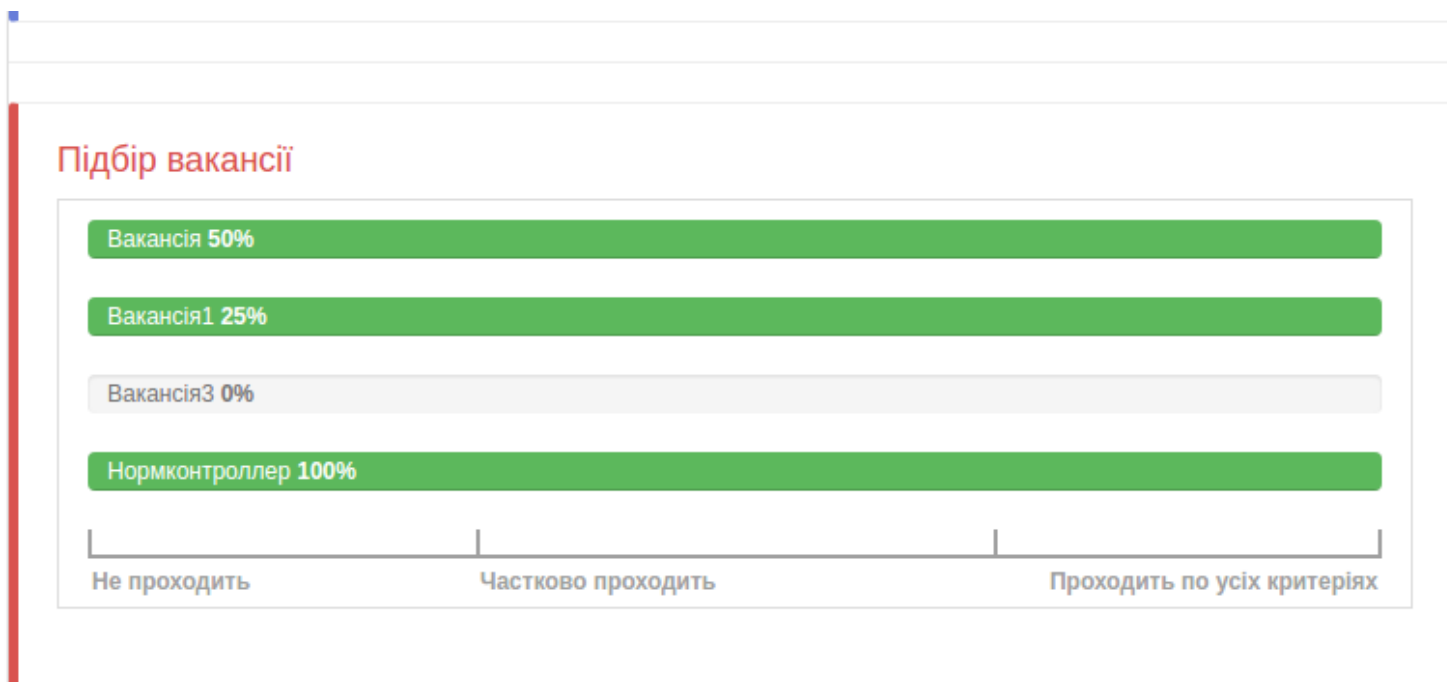


Рисунок 3.10 – відповідність вакансіям

Перейшовши на сторінку вакансій в меню буде показана сторінка всіх вакансій (рисунок 3.11), на сторінці є кнопка переходу до створення вакансії, чи перегляду більш докладної інформації про вакансію (перейти можна аналогічно кандидатам, тобто натиснувши на назву вакансії).

<div> <div>Police Recruiting</div> <div>Новий кандидат</div> <div>Кандидати</div> <div><b>Вакансії</b></div> <div>Статистика</div> <div>Про систему</div> <div>Юрій</div> </div>						
<div> <div>Список вакансій</div> <div> <div>■ - відібрані вакансії</div> <div>■ - вакансії, що проходять відбір</div> <div>■ - не прийняті вакансії</div> <div># - номер вакансії</div> </div> <div>Пошук</div> <div>Створити вакансію</div> </div>						
#	Назва	Категорія	Департамент	Створено	Змінено	Дія
1	Вакансія			2017-05-12 07:21:23	2017-05-12 07:21:23	<div>d</div>
2	Вакансія1			2017-05-12 07:21:34	2017-05-12 07:21:34	<div>d</div>
3	Вакансія3			2017-05-12 07:21:37	2017-05-12 07:21:37	<div>d</div>
4	Нормконтроллер			2017-05-22 09:13:28	2017-05-22 09:13:28	<div>d</div>

© Recrutial Team 2017

Рисунок 3.11 – список вакансій

Після натискання на назву вакансії буде показана більш докладна інформація про вакансію (рисунки 3.12-3.13). Всередині вакансії можна задати важливі поля для вакансії, а також вибрати операцію по якій будуть поля порівнюватися (рисунок 3.14). Поля можна додавати і видаляти і редагувати.

Новий кандидат
Кандидати
Вакансії
Статистика
Про систему

The skills meter from [here](#) is used in this design. Feel free to remove this div from the design.

{освід водіння Водійське посвідчення Робота на ПК Освіта

## Resume of John Doe

© Last Updated on: Sunday, October 05, 2014

300 x 300

John Doe

Software Engineer

Google Inc.

☎ 000-000-0000

✉ john@example.com

### Додавання ваги вакансії

Поле ваги	Операція	Значення
Рекомендації ▾	→ Заповнене ▾	→ 123
Робота на ПК ▾	→ Заповнене ▾	→ 1
ФІО ▾	→ LIKE ▾	→ Юрій

Додати вагу
Зберегти зміни

### Summary

Lorem ipsum dolor sit amet, ea vel prima adhuc, scripta liberavisse ea quo, te vel vidit mollis complectitur. Quis verear mel ne. Munere vituperata vis cu, te pri duis timeam scaevola, nam postea diceret ne. Cum ex quod aliquip mediocritatem, mei habemus persecuti mediocritatem ei.

Odio recteque expetenda eum ea, cu atqui maiestatis cum. Te eum nibh laoreet, case nostrud nusquam an vis. Clita debitis apeirian et sit, integre iudicabit elaboraret duo ex. Nihil causae adipisci id eos.

Рисунок 3.12 – вакансія частина 1

Research Interests

Software Engineering, Machine Learning, Image Processing, Computer Vision, Artificial Neural Networks, Data Science, Evolutionary Algorithms.

Prior Experiences

Software Engineer at Twitter

Lorem ipsum dolor sit amet, ea vel prima adhuc, scripta liberavisse ea quo, te vel vidit mollis complectitur. Quis verear mel ne. Munere vituperata vis cu, te pri dui timeam scaevola, nam postea diceret ne. Cum ex quod aliquip mediocritatem, mei habemus persecuti mediocritatem ei.

Software Engineer at LinkedIn

Lorem ipsum dolor sit amet, ea vel prima adhuc, scripta liberavisse ea quo, te vel vidit mollis complectitur. Quis verear mel ne. Munere vituperata vis cu, te pri dui timeam scaevola, nam postea diceret ne. Cum ex quod aliquip mediocritatem, mei habemus persecuti mediocritatem ei.

- Lorem ipsum dolor sit amet, ea vel prima adhuc, scripta liberavisse ea quo, te vel vidit mollis complectitur. Quis verear mel ne. Munere vituperata vis cu, te pri dui timeam scaevola, nam postea diceret ne. Cum ex quod aliquip mediocritatem, mei habemus persecuti mediocritatem ei.
- Lorem ipsum dolor sit amet, ea vel prima adhuc, scripta liberavisse ea quo, te vel vidit mollis complectitur. Quis verear mel ne. Munere vituperata vis cu, te pri dui timeam scaevola, nam postea diceret ne. Cum ex quod aliquip mediocritatem, mei habemus persecuti mediocritatem ei.

Key Expertise

Lorem ipsum dolor sit amet, ea vel prima adhuc

Lorem ipsum dolor sit amet, ea vel prima adhuc

Lorem ipsum dolor sit amet, ea vel prima adhuc

Lorem ipsum dolor sit amet, ea vel prima adhuc

Lorem ipsum dolor sit amet, ea vel prima adhuc

Lorem ipsum dolor sit amet, ea vel prima adhuc

Education

Degree	Graduation Year	CGPA
Masters in Computer Science and Engineering	2014	3.50
BSc. in Computer Science and Engineering	2011	3.25

Рисунок 3.13 – вакансія частина 2

**Додавання ваги вакансії**

Поле ваги	Операція	Значення
Рекомендації ▼	→ Заповнене ▼	→ 123
Робота на ПК ▼	→ Заповнене ▼	→ 1
ФІО ▼	→ LIKE ▼	→ Юрій

Додати вагу    **Зберегти зміни**

Рисунок 3.14 – важливість пунктів анкети для вакансії

### Висновок

Спираючись на вище описану настанову користувача можна прийти до висновку що система достатньо дружня до користувачів і дозволяє без великих часових витрат навчитися використовувати систему для пошуку та підбору кандидатів . Дизайн розроблено з урахуванням особливостей використання системи і спроектовано таким чином щоб новому користувачу була видна вся найважливіша інформація, а досвідченому забезпечити швидкий доступ до потрібних йому елементів сторінки.

## 4 РОЗРОБКА СТАРТАП-ПРОЕКТУ

### 4.1 Опис ідеї проекту (товару, послуги, технології)

В таблиці 4.1 наведено опис ідеї стартап проекту, напрямки його застосування та вигоди від кожного напрямку для користувача

Таблиця 4.1. Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Розумна система пошуку та відбору персоналу заснована на алгоритмах машинного навчання, нейромережах та підході дистиляції знань.	1. Фільтрація резюме кандидатів	Зменшення кількості резюме які будуть оброблені рекрутерами вручну, можливість використати звільнений час для поглибленого вивчення кандидата
	2. Попередня рекомендація професії кандидата	Надання попередньої оцінки як кандидату так і рекрутеру для визначення найкращої професії кандидата
	3. Ведення обліку вакансій, анкет та резюме кандидатів	Пришвидшення пошуку, категоризації, можливість ведення історії для подальшого аналізу чи навчання системи
	4. Постійне вдосконалення системи за рахунок її навчання.	Підвищення точності визначення професії кандидата з кількістю оброблених резюме

Зміст ідеї	Напрямки застосування	Вигоди для користувача
	5. Можливість отримати відразу навчену систему без заданої спеціалізації	Пришвидшення навчання спеціалізованої системи за рахунок мережі з більшою глибиною, тобто з іншої рекрутингової системи з меншою спеціалізацією

#### 4.1.1 Аналіз потенційних техніко-економічних переваг ідеї

Система має наступних конкурентів:

- workable.com — платана рекрутинова система з великою базою клієнтів;
- gorecru.it — система для працівників hr відділів для пошуку потенційних кандидатів і їх аналізу;
- info.clearcompany.com — мультиплатформне рішення для пошуку та відбору кандидатів.

В таблиці 4.2 нижче наведено порівняння та визначення характеристик ідеї проекту

Таблиця 4.2. Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№ п/п	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W (слабкі сторони)	N (нейтральні сторони)	S (сильні сторони)
		Мій проект	workable.com	gorecru.it	info.clearcompany.com			
1	Вартість обслуговування	Відносно невелика	Не знайдено	Не знайдено	Не знайдено		+	

№ п/п	Техніко- економічні характерист ики ідеї	(потенційні) товари/концепції конкурентів				W (слабк а сторо на)	N (нейт ральн а сторо на)	S (силь на сторо на)
		Мій проект	workabl e.com	gorecru .it	info.cle arcomp any.co m			
			інформ ації	інформ ації	інформ ації			
2	Вартість експлуатаці ї	20\$ за рекрутера в місяць + 100\$ за платну підписку	Не знайде но інформ ації	Не знайде но інформ ації	Не знайде но інформ ації		+	
3	Безвідмовні сть	Завдяки мікросервіс ній архітектурі шанси на виведення з ладу невеликі, також є можливість швидко відновити роботу у випадку експередбаче	Досить висока	Дуже висока	Достат ня		+	



№ п/п	Техніко- економічні характерист ики ідеї	(потенційні) товари/концепції конкурентів				W (слабк а сторо на)	N (нейт ральн а сторо на)	S (силь на сторо на)
		Мій проект	workabl e.com	gorecru .it	info.cle arcomp any.co m			
		них обставин						
4	Ремонтопри датність	Завдяки мікросервіс ній архітектури необхідно ремонтувал и лише окремий сервіс	Не знайде но інформ ації про архітек туру	Не знайде но інформ ації	Не знайде но інформ ації		+	
5	Оплата праці	За рахунок того, що система є стартапом. Спочатку система буде розвиватись завдяки ентузіазму команди	Дуже висока	Висока	Висока			+

№ п/п	Техніко- економічні характерист ики ідеї	(потенційні) товари/концепції конкурентів				W (слабк а сторо на)	N (нейт ральн а сторо на)	S (силь на сторо на)
		Мій проект	workabl e.com	gorecru .it	info.cle arcomp any.co m			
6	Зручність користуван ня	Сучасні та зручні UI/UX рішення	Низька	Середн я	Висока			+
7	Простота освоєння	Сучасні та зручні UI/UX рішення	Сердня	Середн я	Висока так як сай існує давно і швидш е за все було провед ено достат ньо дослід жень для його покра щення)		+	

№ п/п	Техніко- економічні характерист ики ідеї	(потенційні) товари/концепції конкурентів				W (слабк а сторо на)	N (нейт ральн а сторо на)	S (силь на сторо на)
		Мій проект	workabl e.com	gorecru .it	info.cle arcomp any.co m			
8	Дизайн ресурсу	Середній	Середні й	Середн ій	Середн ій		+	
9	Відповідніс ть патернами дизайну	Висока	Низька	Середн я	Низька			+
10	Безпека даних користувач а	Висока, за рахунок того, що це одна з основних вимог до системи	Дуже висока	Дуже висока	Середн я		+	
11	Відповідніс ть тенденціям дизайну	Висока	Висока	Дуже висока	Середн я	+		
12	Відмово стійкість системи	За рахунок багаторівне вої архітектури та бекапів даних дуже	Дуже висока	Дуже висока	Середн я		+	

№ п/п	Техніко- економічні характерист ики ідеї	(потенційні) товари/концепції конкурентів				W (слабк а сторо на)	N (нейт ральн а сторо на)	S (силь на сторо на)
		Мій проект	workabl e.com	gorecru .it	info.cle arcomp any.co m			
		мала ймовірність виходу всієї системи з ладу та втрати якихось даних						

#### 4.2 Технологічний аудит ідеї проекту

В таблиці 4.3 наведено обґрунтування технологічної здійсненності проекту

Таблиця 4.3. Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології реалізації	Наявність технологій	Доступність технологій
1	Виконання процесу навчання за рахунок фільтрації вже готових знань	Аналогічні системи або системи створенні компанією	Потребує розробки та впровадження	Доступна

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
2	Фільтрація поступаючих резюме кандидатів	Використання розроблених алгоритмів , а також накопичених ними «знань»	Потребує розробки	Використовуються власні розробки
3	Визначення ймовірної професії претендента	Використання розроблених алгоритмів , а також накопичених ними «знань»	Потребує розробки	Використовуються власні розробки
4	Ведення бази даних резюме , кандидатів, анкет та рекрутерів	Використання розробленого застосунку	Потребує розробки	Використовуються власні розробки
5	Інформаційна система	Використання засобів розробки	Потребує розробки	Використовуються засоби, що є вільними для використання
Обрана технологія реалізації ідеї проекту: Фільтрації резюме на першому кроці пошуку кандидата підбір вакансії на подальших кроках та ведення поточного та минулого обліку даних зв'язаних з роботою рекрутингового відділу.				

### 4.3 Аналіз ринкових можливостей запуску стартап-проекту

Нижче наведена таблиця 4.4, в якій описуються попередні характеристики потенційного ринку проекту. В таблиці наведені ідеї проекту та характеристики технологій їх реалізації

Таблиця 4.4. Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Збір даних про поточну ситуацію на ринку та в компанії	Поточна БД застосунку та використання припущень алгоритмів машинного навчання та нейронних мереж	Потребує розробки	Доступна
2	Підбір професії чи вакансії	Використання алгоритмів машинного навчання	Потребує розробки	Використовуються власні розробки
3	Фільтрація резюме	Використання алгоритмів машинного навчання	Потребує розробки	Використовуються власні розробки
4	Інформаційна система	Використання засобів розробки	Потребує розробки	Використовуються засоби, що є вільними для використання

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
Обрана технологія реалізації ідеї проекту: Фільтрації режюме на першому кроці пошуку кандидата підбір вакансії на подальших кроках та ведення поточного та минулого обліку даних зв'язаних з роботою рекрутингового відділу.				

В таблиці 4.5 характеризуються потенційні клієнти стартап проекту, наведені потреби цільової аудиторії та особливості її поведінки

Таблиця 4.5. Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
1	Важливість підбору кваліфікованих чи підходящих за рядом ознак кандидатів до відповідних їм вакансій	Поточні чи колишні рекрутери, люди на ринку праці	Загальні поведінкові патерни схожі з невеликими відмінностями в залежності від бажаної професії	Висока точність підбору кандидата до вакансії та висока точність рекомендації професії кандидата
2.	Зручність використання системи	Всі користувачі системи	Всі групи бажають швидко освоюватися в системі та швидко виконувати операції в ній	Зручний UI/UX системи зокрема частини веб

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
				застосунку для кандидатів
3.	Висока відмовостійкість системи	Всі користувачі системи	Перш за все потреба в відмовостійкості виникає у рекрутерів, так як саме вони зацікавлені в отриманні висококваліфікованого чи легконавчуваного кадра	Система повинна бути працездатною в будь який час

Також в таблиці 4.6 оцінюються фактори загроз для стартап проекту

Таблиця 4.6. Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Сильні конкуренти	Поточні конкуренти знають і вміють використовувати досвід та великі бази даних кандидатів	Маркетингова кампанія, використання системи з алгоритмами машинного навчання для того щоб



№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
			запропонувати користувачам більше ніж в ніж є в конкурентів
2	Низька зацікавленість користувачів	Недовіра користувачів до нової системи	Реалізація якісного з конкурентними UI/UX продукту та маркетингова кампанія
3	Швидкі зміни на ринку праці	Поточні вимоги та пропозиції на ринку праці швидко змінюються	Частіше перенавчати систему

Окрім цього в таблиці 4.7 наведені можливості які має стартап проект та їх обґрунтування.

Таблиця 4.7. Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Отримання корисного фідбеку від «перших» користувачів системи	Покращення продукту на основі реального досвіду використання системи	Аналіз та врахування відгуків

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
2	Впровадження нового функціоналу на основі даних про користувачів	Розширення системи	Побудова плану розвитку системи

Ступеневий аналіз поточних конкурентів на ринку пошуку та відбору персоналу наведено у таблиці 4.8.

Таблиця 4.8. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - монополія/олігополія/ монополістична/чиста	Чиста, так як кількість ресурсів котрі задовольняють потреби як працедавців так і кандидатів досить велика	Мати свою власну «фішку» яка відрізнятиме систему від аналогів та корисність «фішки»
2. За рівнем конкурентної боротьби - локальний/національний/...	Національний так як роботу шукають зазвичай у рамках країни проживання	Немає
3. За галузевою ознакою	Внутрішньогалузева, так як всі є інтернет ресурсами	Моніторинг та використання останніх тенденцій

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
4. За характером конкурентних переваг - цінова / нецінова	Цінова, так як для рекрутингових агенств та рекрутингових відділів компаній система платна	Підвищення якості системи, використання підходів та методик котрі зменшують затрати на підтримання системи

Обґрунтування конкурентоспроможності наведено в таблиці 4.9.

Таблиця 4.9. Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Використання методів машинного навчання для рекомендацій майбутньої професії	Зменшує кількість людською роботи а також «людський фактор», як наприклад корупція, помилки, некомпетентність чи лень
2	Можливість майже моментально навчити систему	Методом початкового навчання системи є підхід teacher-student, завдяки

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
		якому початкове навчання системи проходить швидко
3	Висока надійність системи	Завдяки мікросервісній архітектурі виведення всієї системи з ладу мало ймовірна, а при правильно налаштованих системах резервного копіювання та розташування програми окрім того що шанс поломки малий навіть якщо вона станеться то буде можливість розгорнути аналогічний компонент системи досить швидко
4	Підсистема ведення обліку кандидатів, вакансій, рекрутерів та операцій між ними	Створення та накопичення свого масиву даних (анкет, резюме, просто імен та професій кандидатів) є важливою та досить дорогою інформацією яку кожна компанія бажає

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проєктів значущим)
		бачити під своїм контролем
5	Закладена можливість розширення системи	Система спроектована так щоб можна було легко додати нові компоненти або з мінімальними ресурсними та часовими затратами видалити старі.

### Результати порівняння слабких та сильних сторін

Таблиця 4.10. Порівняльний аналіз сильних та слабких сторін системою пошуку та підбору персоналу

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з системою пошуку та підбору персоналу							
			-3	-2	-1	0	+1	+2	+3	
1	Використання методів машинного навчання для рекомендацій майбутньої професії	20		+						

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з системою пошуку та підбору персоналу						
			-3	-2	-1	0	+1	+2	+3
2	Можливість майже ментально навчити систему	20	+						
3	Висока надійність системи	15						+	
4	Підсистема ведення обліку кандидатів, вакансій, рекрутерів та операцій між ними	10						+	
5	Закладена можливість розширення системи	10				+			

SWOT- аналіз стартап-проекту наведений в таблиці 4.11.

Таблиця 4.11. SWOT- аналіз стартап-проекту

Сильні сторони:	Слабкі сторони:
Використання алгоритмів машинного навчання та нейронних мереж Можливість швидко навчити спеціалізовану аналогічну систему Можливість розширення системи Висока надійність	Слабкий дизайн У випадку масштабної модифікації потрібні спеціалізовані кадри
Можливості: Фідбек від реальних користувачів	Загрози: Велика кількість конкурентів

Сильні сторони:	Слабкі сторони:
Оптимізація всього процесу рекрутингу Зменшення впливу «людського фактору»	Низька зацікавленість користувачів

Можливості альтернативного впровадження на ринку для випадку коли основна стратегія не діє наведено в таблиці 4.12

Таблиця 4.12. Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Виведення на ринок робочої системи без використання алгоритмів машинного навчання та нейронних мереж	Ресурси будуть отримані від користувачів котрі зацікавляться майбутніми модифікаціями системи	4 місяці
2	Проведення досліджень та покращення UI/UX на основі реальних даних від реальних користувачів	Ресурси від збільшення кількості клієнтів	1 місяць
3	Відміна платної підписки та введення	Отримані від реклами	2 місяці

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
	реклами для веб застосунку		

#### 4.4 Розроблення ринкової стратегії проекту

Було визначено кілька цільових потенційних груп споживачів котрих може зацікавити стартап проект. Їх опис та певні характеристики наведені в таблиці 4.14.

Таблиця 4.14. Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Люди котрі шукають роботу	Висока	Високий так як людина котра шукає роботу використовує різноманітні джерела інформації про вакансії	Висока	Середня тому що окрім високої конкуренції у системі є переваги
2	Рекрутери компаній	Висока	Високий так як продукт	Середня	Середня так як хоча рішення і



№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
			достатньо зменшує обсяги робіт які має виконувати людина		скорочує час та ресурси використані на підбір кандидатів впровадження цього рішення може спіткнутися об застарілі процеси компаній, а у випадку з великими компаніями що давно на ринку таке радіше правило ніж виняток
3	Рекрутингові агенства	Середня так як віддавати свої бази	Середній	Низька	Середня так як не зважаючи на те що конкуренції в

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
		даних в побічну систему потребує достатньої довіри до виробника системи			агрегації даних з різних рекрутингових не так багато але і відсутня вона не без причини – ці дані приносять прибуток агенціям і тому просто так вони ними неподіляться
Які цільові групи обрано: люди котрі шукають роботу, рекрутери компаній, рекрутингові агенства					

Була створена базова стратегія розвитку стартап проекту, а також визначені конкурентоспроможні позиції проекту. Стислу інформації про неї можна побачити в таблиці 4.15

Таблиця 4.15. Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
1	Впровадження системи лише як продукту для кандидатів без впровадження у роботодавців як окрема система	Використання безкоштовної моделі розповсюдження охоплення найбільш масової аудиторії	Автоматичні рекомендації вакансій кандидатам і їх пріорітизація	

Були досліджені поведінкові стратегії конкурентів та на їх основі визначена базова стратегія конкурентної поведінки на ринку.

Таблиця 4.16. Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	Не є «першопрохідцем»	Буде забирати існуючих споживачів у конкурентів	Частина буде аналогічною, а частина унікальною (використання нейронних мереж для рекомендацій)	Виведення продукту на ринок і пропозиція нової функціональності відносно того що є в конкурентів.

В таблиці 4.17 наведені основні пункти стратегії позиціонування

Таблиця 4.17. Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартаппроекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Надійність системи	Використання мікросервісної архітектури системи для підвищення надійності	Збільшення надійності системи та простоти розширення та масштабування	Надійність, простота, масштабованість
2	Точність прогнозів	Пришвидшення навчання нейромережі використанням методу «дистиляції знань»	Система здатна сама навчатися та аналізувати дані, до того ж може початкове навчання робити досить швидко	Швидкість, точність, незалежність

#### 4.5 Розроблення маркетингової програми стартап-проекту

Ключові переваги продукту стартап проекту описані в таблиці 4.18.

Таблиця 4.18. Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Скорочення витрат часових ресурсів	Фільтрація вхідних резюме, рекомендації вакансій та кандидатів – оптимізація всього процесу рекрутингу	Самонавчання системи, можливість підвищувати точність прогнозів з часом, сама можливість рекомендації
2	Самовизначення професії кандидата	Автоматична рекомендація вакансій кандидату на основі анкетних даних	Рекомендація вакансій

Цінова політика стартапу стисло описана в таблиці 4.19

Таблиця 4.19. Визначення меж встановлення ціни

№ п/п	Рівень цін на товаризамінники	Рівень цін на товарианалоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	20\$ за користувача рекрутера на місяць, кандидат безкоштовно	Невідомо	Достатньо високий для оплати кожного рекрутера	Нижня – безкоштовне користування Верхня – 100\$ за користувача рекрутера

Особливості системи збуту наведені в таблиці 4.20

Таблиця 4.20. Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Клієнт самостійно купує та використовує продукт, виробник виконує	Збут власними силами	Висока	Платна підписка

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
	робочу підтримку			

Концепція маркетингових комунікацій стартап проекту системи пошуку та підбору персоналу наведено в таблиці 4.21.

Таблиця 4.21. Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Клієнти готові віддавати гроші за якісний та діючий продукт і достатньо добра розбираються в цільовій функціональності аби розуміти	Інтернет	Інформаційна система пошуку та відбору персоналу	Через мережу інтернет чи таргетовану рекламу	Використання таргетованої реклами



№ п/ п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуютьс я цільові клієнти	Ключові позиції, обрані для позиціонуванн я	Завдання рекламного повідомленн я	Концепція рекламного звернення
	ефективність продукту				

### Висновок

З вище наведеного можна зробити висновок що «Інформаційна система пошуку та відбору персоналу» є достатньо ефективним стартапом. Після аналізу конкуренції з аналогічними системами може вважати що поточний стартап зможе хоч і не зразу але бути суперником поточним лідерам ринку. Біли розроблені альтернативні стратегії розвитку для використання при можливих ускладненнях чи несподіваних змінах на ринку. Систему розповсюдження за підпискою можна вважати досить поширеною для такого типу систем тому її ефективність є достатньою для впровадження в систему.

## ВИСНОВКИ

Метою цієї магістерської дисертації було пришвидшити процес підбору персоналу використовуючи підходи машинного навчання та алгоритми нейронних мереж.

Для отримання даних про поточних кандидатів було створено систему котра дозволяє додавати, редагувати та змінювати надані ними анкети та резюме. Аналогічно в системі була реалізована можливість управління вакансіями для того щоб рекрутери могли сповіщати кандидатів про необхідність зайняття робочого місця.

Звісно окрім можливості зберігати та редагувати інформацію про кандидатів, вакансії, важливість певних пунктів вакансії система використовує алгоритми машинного навчання для фільтрації резюме та рекомендацій кандидатам вакансій і навпаки. Для отримання такого результату було використано декілька підходів та алгоритмів. Першим є метод опорних векторів - один з класичних алгоритмів машинного навчання котрий дозволяє прогнозувати професію кандидата за його анкетною, а також підбирати найкращого кандидата до вакансії. Для порівняння та покращення результатів було використано ще один алгоритм машинного навчання та проведена його модифікація.

Іншим використаним алгоритмом є згорткова нейронна мережа, котра виконує аналогічні методу опорних векторів цілі. Згорткова нейронна мережа має більший потенціал для навчання але сама по собі не дає великого виграшу у порівнянні з методом опорних векторів у рамках поставленої мети дисертації.

Тому було прийнято рішення використати методі teacher-student для навчання мережі, котра використовувалася для нашої задачі, мережею з більшою глибиною та вже існуючими знаннями. Метод дозволив фактично виключити крок навчання мережі, так як знання напряду переносяться з більш глибокої мережі в менш глибоку – нашу.

В подальшому було прийнято рішення збільшити точність навченої мережі без втрати в швидкості. Для цього була розроблена модифікація методу заснована на гаусівських шумах. Новий метод дозволяє імітувати навчання у декількох учителів що збільшує точність навченої мережі без значних втрат в швидкості навчання.

Результати яких вдалося досягти завдяки модифікації наведено в розділі 2, і з них можна зробити висновок що розроблений метод працює та зашумлення знань дійсно зменшує кількість помилок котрі допускає мережа. Але не менш важливим є те що рівень шуму в даних потрібно підбирати з розумінням самого процесу так як видно з другого експерименту при меншій кількості даних занадто високий рівень шуму може погіршувати точність результатів.

## ПЕРЕЛІК ПОСИЛАНЬ

1. J. Ba and R. Caruana. Do deep nets really need to be deep? In Advances in neural information processing systems, 2014. (2654-2662)
2. G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. arXiv preprint, 2015.
3. C. M. Bishop. Training with noise is equivalent to tikhonov regularization. Neural computation, 1995. (108-116)
4. C. Bucilu, R. Caruana, and A. Niculescu-Mizil. Model compression. In Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2006. (535-541)
5. W. Chen, J. T. Wilson, S. Tyree, K. Q. Weinberger, and Y. Chen. Compressing neural networks with the hashing trick. CoRR, 2015.
6. M. Denil, B. Shakibi, L. Dinh, N. de Freitas, et al. Predicting parameters in deep learning. In Advances in Neural Information Processing Systems, 2013.(2148-2156)
7. Y. Gong, L. Liu, M. Yang, and L. Bourdev. Compressing deep convolutional networks using vector quantization.
8. G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. arXiv preprint, 2015.
9. <http://www.mnis.fr/> - open source solutions for network training
10. <http://ufldl.stanford.edu/housenumbers/> - open source solutions for network training
11. [https://pidruchniki.com/13340203/menedzhment/vidbir\\_personalu](https://pidruchniki.com/13340203/menedzhment/vidbir_personalu) ресурс Відбір та управління персоналом
12. Talent Management Software with a Clear Mission for Success [Електронний ресурс] Режим доступу: <http://info.clearcompany.com/>
13. SVM [Електронний ресурс] Режим доступу: <http://crypto.pp.ua/2011/05/metod-opornyx-vektorov/>

**ДОДАТОК А**  
**ГРАФІЧНИЙ МАТЕРІАЛ**

**ПЛАКАТ 1. ДІАГРАМА ВАРІАНТІВ ВИКОРИСТАННЯ**

**ПЛАКАТ 2. ДІАГРАМА РОЗГОРТАННЯ**

**ПЛАКАТ 3. ДІАГРАМА ДІЯЛЬНОСТІ**



**ПЛАКАТ 4. ER-ДІАГРАМА**

**ПЛАКАТ 5. ДІАГРАМА КЛАСІВ**

**ПЛАКАТ 6. ДІАГРАМА КОМПОНЕНТІВ ЗАСТОСУВАННЯ**

**ПЛАКАТ 7. ДІАГРАМА ПОСЛІДОВНОСТІ**